A Baseline for Shapely Values in MLPs: from Missingness to Neutrality

Cosimo Izzo
University College London,
Now-casting Economics
cosimo.izzo.18@ucl.ac.uk

Aldo Lipani
University College London
aldo.lipani@ucl.ac.uk

Ramin Okhrati
University College London
r.okhrati@ucl.ac.uk

Francesca Medda
University College London
f.medda@ucl.ac.uk

Abstract

Being able to explain a prediction as well as having a model that performs well are paramount in many machine learning applications. Deep neural networks have gained momentum recently on the basis of their accuracy, however these are often criticised to be black-boxes. Many authors have focused on proposing methods to explain their predictions. Among these explainability methods, *feature attribution methods* have been favoured for their strong theoretical foundation: the Shapley value. A limitation of Shapley value is the need to define a baseline (aka reference point) representing the *missingness* of a feature. In this paper, we present a method to choose a baseline based on a *neutrality* value: a parameter defined by decision makers at which their choices are determined by the returned value of the model being either below or above it. Based on this concept, we theoretically justify these neutral baselines and find a way to identify them for MLPs. Then, we experimentally demonstrate that for a binary classification task, using a synthetic dataset and a dataset coming from the financial domain, the proposed baselines outperform, in terms of local explanability power, standard ways of choosing them.

1 Introduction

Nowadays, with a growing interest in modelling problems with deep learning, decision makers have sometime shown reluctance in accepting the prediction made by these models because perceived as black-boxes. Indeed, in disciplines such as economics, finance and healthcare, the ability to explain predictions is as important as having a model that performs well (Caruana et al., 2015; Tsang et al., 2018; Goodman and Flaxman, 2017).

A solution to the problem is provided by feature attribution methods, which are used to indicate how much each feature contributes to the prediction for a given example. A theoretically grounded feature attribution method is provided by Shapley value and its approximations (Lundberg and Lee, 2017; Ancona et al., 2019). When explaining a prediction with Shapely value, we need to perform two steps. First we need to define a *baseline*, then given an example, we compute the Shapely value of each of its feature. While most works focused on the latter, the former has not been sufficiently explored, although the definition of the baseline is fundamental in order to correctly interpret Shapely values (Kindermans et al., 2019).

The common practice in choosing a baseline is to use the vector of zeros as a baseline (Zeiler and Fergus, 2014; Shrikumar et al., 2017; Sundararajan et al., 2017; Ancona et al., 2019), which coincides

Preprint. Under review.

with the average baseline when features are standardised. However, such a generic choice does not fit all applications. For example, in a classification task, with binary features representing the presence or absence of an entity, given a example and its prediction value, such a baseline would always measure a null contribution for each feature with value equal to zero.

Ill-defined baselines can drastically change the interpretation of these Shapely values. The use of improper baselines causes issues when interpreting the relative importance of features. In particular, most of the baselines proposed for attribution methods are selected independently with respect to the model. Thus, using the example of a classification problem, they do not explain the prediction of the model, but rather a difference with respect to some arbitrary value taken by the model.

In this paper we aim to solve this issue by introducing the concept of *neutrality value*. This value is used as a parameter and it represents the point at which a user of an ML model is not sure about whether to take a decision rather than another. This, for example, could be the probability of default at which a bank decides to approve or reject a loan request. Based on this concept, we theoretically demonstrate the existence of at least a *neutral baseline* based on a dataset and a model. Here, we develop an algorithm to quickly search for this baseline. Using a synthetic dataset and a real dataset coming from the financial domain, we then empirically demonstrate the ability of this baseline to have higher explanability power than other choices of baseline.

The reminder of the paper goes as follows: § 2 introduces the background and related work about Shapely value, standard choice of baselines, and how feature attribution methods are evaluated; § 3 introduces our method; § 4 presents the experiments and results, and; we conclude in § 5.

2 Background and Related Work

2.1 Shapley Values, Their Approximations, and Baselines

Applications of Shapley values for explainability in ML have been introduced by Lundberg and Lee (2017). For a neural network G_θ with parameters θ, and K input features, the contribution of feature j calculated according to the Shapely value for input $x = [x_1, x_2, \ldots, x_K]$ is given by:

$$\sum_{S \subseteq P \setminus \{j\}} \frac{|S|!(|P| - |S| - 1)!}{|P|!} \left(G_\theta(\tilde{x}_{S \cup \{j\}}) - G_\theta(\tilde{x}_S) \right),$$

where P is the collection of all feature indexes, element i of vector $\tilde{x}_S$ is given by $\tilde{x}_{S,i} = x_i \mathbb{1}_{\{i \in S\}} + b_i \mathbb{1}_{\{i \notin S\}}$ (similarly for $\tilde{x}_{S \cup \{j\}}$), and b_i is the baseline value for feature i. The *baseline* models the missingness of a feature, i.e., it replaces that feature when it is absent. As it is argued by Sturmfels et al. (2020), unlike game theory, the concept of missingness is not well defined and explored in ML models. The standard practice in setting up a baseline is to assign a vector of zeros for all features. However, this choice might provide wrong interpretations and even give zero importance to significant features.

On the other hand, as the number of features increases, calculating Shapley values becomes expensive as the computational cost is exponential with respect to the number of features. Hence, estimation methods have been developed, and the most commonly used ones in Deep Learning are Deep Taylor Decomposition (Montavon et al., 2017), Integrated Gradient (Sundararajan et al., 2017), Deep SHAP (Lundberg and Lee, 2017), and Deep LIFT (Shrikumar et al., 2017). Other model agnostic methods are: Sampled and Kernel SHAP (Lundberg and Lee, 2017), and LIME (Ribeiro et al., 2016). All of these methods require the choice of a baseline. In the next section, we review commonly used methods in choosing a baseline.

2.2 The Choice of a Baseline

As mentioned earlier, the simplest choice of a baseline is the **zero** vector baseline (Zeiler and Fergus, 2014; Sundararajan et al., 2017; Shrikumar et al., 2017; Ancona et al., 2019). However, this choice could be misleading. One way of addressing the zero-baseline insensitivity problem is to use the maximum distance baseline (**mdb**) method as in Sturmfels et al. (2020). This baseline consists in taking the furthest observation from the current one in an L^1 norm. However, this approach unequivocally creates incoherent justifications for the interpretations provided by the model due to the high correlation of the baseline with the underlying dataset.

Alternatively, one can sample a baseline from a multivariate distribution such as uniform (Sturmfels et al., 2020) or Gaussian (Smilkov et al., 2017). This approach can be improved by considering a sample of baselines and to **average** the attributions computed over each baseline (Smilkov et al., 2017; Lundberg and Lee, 2017; Erion et al., 2019; Kapishnikov et al., 2019; Sundararajan and Najmi, 2019). Another form of sampling is the one performed using the dataset. Hence, one can use the underlying empirical distributions of the dataset (Sturmfels et al., 2020). We denote this as the $\mathbf{p}_X$ baseline method. However, both $\mathbf{p}_X$ and **average** baselines increase the computational cost of estimating feature attributions linearly with respect to the number of draws. Further, the expectation over a run on multiple baselines could be a reasonable option to compute global feature importance, yet it is difficult to envisage the concept of missingness in such an approach.

For the Deep Taylor Decomposition approach, the baselines are chosen using Layer-wise Relevance Propagation (Bach et al., 2015) or Pattern Attribution (Kindermans et al., 2017). The drawback of these baseline search methods is the non applicability to other attribution methods. For this reason we do not compare against them.

2.3 Measure of Explainability Power

The evaluation of explainability methods from a quantitative prospective is difficult due to the lack of a clear definition of what is a correct explanation (Samek et al., 2016; Sundararajan et al., 2017; Ancona et al., 2019). Many extrinsic evaluation of explanability power have been developed. These are based on measuring the effect on a model when removing the most important feature as identified by the feature attribution methods. Some quantify this effect by measuring the difference in performance, and others by measuring the difference in prediction value. The intuition behind these methods is that if the feature attribution method correctly identifies the most important feature, then when this feature is removed, the model performance should decrease more (or the prediction value should deviate more) than when a less important feature is removed. In this paper, we use these evaluation methods to compare the choice of baselines.

These evaluation methods can be applied *globally* or *locally*. These are global when the discovery of the most important feature is done by analysing a large sample of data points. While, these methods are local when the discovery of the most important feature is done on a per example basis. Hooker et al. (2019) propose a global measure called RemOve And Retrain (ROAR). This consists in, given a model, to first identify the most important feature using the whole dataset, then retrain the model without this feature and measure the difference in performance between the two models. Ancona et al. (2017) proposed a measure of local evaluation which consists in, given a model, to first identify the most important feature for each example, then removing this feature by substituting it with its baseline value and measure the deviation in prediction. These deviations are then averaged to get a single value. The limitation of these evaluation methods is that since a feature is removed once at the time, these measures may be mislead by potential feature correlations. In this paper, to avoid this issue, we also perform an analysis using a synthetic dataset where features are generated independently.

3 The Neutral Baseline

In this section, we theoretically justify the existence of a baseline according to a well defined concept of neutrality value and present the algorithm to find these baselines in MLPs.

The intuition behind this baseline is better explained by the following motivating example. Consider a classifier used by a banking agent to predict the probability of default of clients based on a set of features. Assume that, the classifier has been trained with standardised features, these features have different importance, and the decision boundary of the classifier does not cross the origin. Imagine now that a client, whose feature values are equal to an average clients' feature values, complains about a rejected loan application, and the banking agent wants to explain the decision of the classifier using Shapely values with a vector of zeros as a baseline. In this case, the agent is not going to be able to provide a meaningful explanation since the Shapely values will all be equal to 0 (due to the standardisation of the features and the given baseline). However, if the baseline is chosen on the decision boundary of the classifier, the Shapley values, so calculated, will be equal to the correct relative importance of the features, making the agent able to explain the prediction. Although, this is about a client with very specific feature values, a more detailed mathematical analysis of this problem

shows that misleading attributions of importance happens every time the client's feature values lie between the baseline of zeros and the decision boundary of the classifier.

In this paper, we argue that the baseline should live on the decision boundary of the classifier, this because, going back to the example above, if the baseline was on the decision boundary, then the Shapely values would have been different from zero and with positive values for those features causing the loan to be rejected. We relate the concept of missingness in Shapely values to the output of the model through the following definition:

Definition 1 (Neutrality Value)**.** *Given a model prediction $\hat{y}$ and a decision maker, we say that the value a is neutral if the decision maker's choice is determined by the value of $\hat{y}$ being either below or above a.*

This neutrality value is usually set by the decision maker. For example, consider a probabilistic classifier used in loan issuing in the motivating example above. Suppose that the banking agent is happy to approve a loan if the client has less than a 50% chance of default, then the neutrality value should be set to $a = 0.5$. Hypothetically, if the model's outputs is 0.5, then the agent will be indecisive between issuing the loan or not.

The idea is that this neutrality value can lead to a point in the input domain that could be used as a baseline. However, given a neutrality value and an SLP with more than one input feature, there are an infinite number of possible combinations of such inputs that lead to the same neutral output. Therefore, we narrow down the solutions to the ones that are fair with respect to the features: we say that x is in the space of *fair* baselines if every element x_i, when sorted with respect to all of its possible realisations, has the same number of such realisations to its left or right with respect to the left or right of any other element of x. We call this fair baselines because we are being fair in representing each feature in the input space. The following definition formalises this:

Definition 2 (Space of Fair Baselines)**.** *Consider a dataset in $\mathbb{R}^k$, $k \geq 1$, generated by a distribution. A vector $x \in \mathbb{R}^k$ is in the space of fair baselines (hence called a fair baseline) if*

$$\max[C_j(x_j), 1 - C_j(x_j)] = \max[C_i(x_i), 1 - C_i(x_i)], \tag{1}$$

where C_is are the marginal CDFs, and the equality holds for any two C_i and C_j and the corresponding elements x_i and x_j of x.

Based on the two definitions, neutrality value and space of fair baselines, in what follows, we demonstrate the existence of a fair baseline that when given to an MLP returns the neutrality value. Before doing this, we need to state the following two assumptions:

A1. All activation functions are monotonic and continuous.

A2. All marginals cumulative distribution functions (CDFs) of the joint CDF of the input features are bijective and continuous.

Assumption **A1** is reasonable since many activation functions are monotonic like linear, sigmoid, tanh, softplus, ReLU, LeakyReLU and ELU (both with parameter $\alpha > 0$). All of these functions are continuous. **A2** is instead a technical assumption needed in the proof of our result later.

3.1 The SLP Case

Using Definitions 1 and 2, **A1** and **A2**, the following proposition guarantees the existence of a fair baseline for SLPs:

Proposition 1 (Existence of a Fair Baseline for SLPs)**.** *Given an SLP (G_θ) satisfying **A1**, a dataset satisfying **A2**, and a neutrality value a in the image of G_θ, then there exists at least a fair baseline x such that $G_\theta(x) = a$.*

The proof of Proposition 1 (provided in Appendix 1) suggests a way to find one fair baseline for an SLP. This is formalised in Algorithm 1. This algorithm using empirical CDFs instead of theoretical ones, requires as inputs an SLP (G_θ), a neutrality value (a), a quantile function (Q_j, obtained from the marginal empirical CDF) for each dimension of input features, a granularity level $\delta > 0$, and a tolerance level $\epsilon > 0$. The δ controls the speed of search, the ϵ controls the margin of error in finding a baseline such that $G_\theta(x) = a$. This algorithm starts the search from the lowest possible fair baseline which is when $p = 0$, and it stops when it reaches a point which is close enough to a. This is

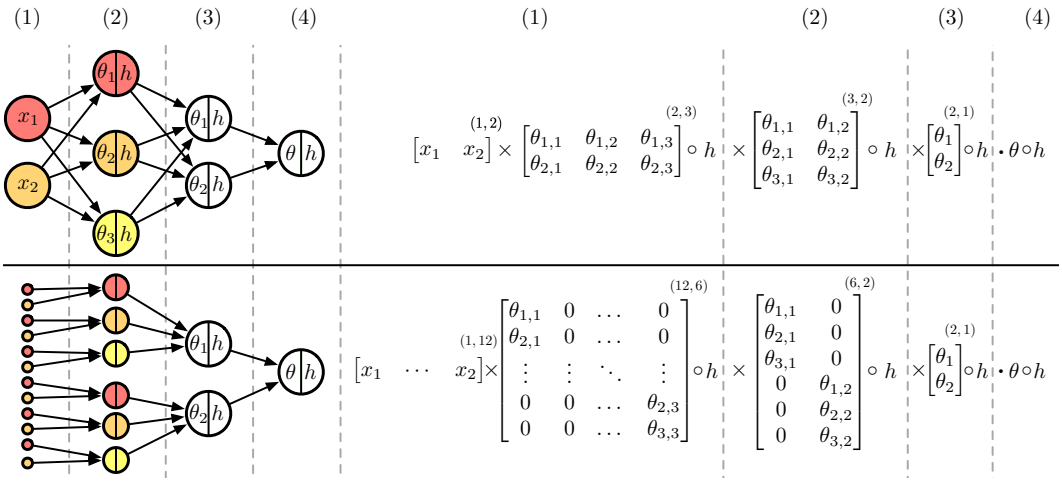


Figure 1: An MLP (above) and its equivalent sparse representation (below).

possible because using the parameters of the SLP we can restrict and define an order in function of p for the set of fair baselines (as in Line 2). This allows us to test these baselines from the smallest SLP value to the largest.

Algorithm 1 Neutral Baseline Search for SLPs

Input: $G_\theta, a, Q, \delta, \epsilon$
Ouput: b

1: $p \leftarrow 0$
2: $b \leftarrow [Q_j(\mathbb{1}_{\theta_j>0} \cdot p + \mathbb{1}_{\theta_j<0} \cdot (1-p)) \; \forall j \in 1, .., K]$
3: **while** $G_\theta(b) - a < \epsilon$ **do**
4: $\quad p \leftarrow p + \delta$
5: $\quad b \leftarrow [Q_j(\mathbb{1}_{\theta_j>0} \cdot p + \mathbb{1}_{\theta_j<0} \cdot (1-p)) \; \forall j = 1, .., K]$
6: **end while**

3.2 The MLP Case

Finding a baseline for MLPs is more complicated, because there is no easy way to order the baselines in function of p as in the SLP case.

We observe that an MLP with L layers can be decomposed into $\sum_{l=2}^{L} \prod_{l'=l}^{L} k_{l'}$ SLPs. This is done by replicating every node at layer l, k_{l+1} times, i.e., the number of nodes at layer $l+1$, and considering every node in the layer $l+1$ as an SLP with input given by the layer l. Based on this observation, we can recursively apply Alg. 1 backwards through the layers of the model to recover the neutrality values across those SLPs, from the output layer to the input layer. This will provide $\prod_{l=2}^{L} k_l$ baselines, one for each SLP in the first hidden layer. This is implemented in Alg. 2. Note the concatenation of the found fair baselines from Line 13 to 16.

Finally, in order to aggregate these baselines, we define an equivalent sparse representation of an MLP (SparseMLP), which is constructed by concatenating each of the SLPs defined above. An example of this transformation is provided in Figure 1. We find this necessary because to compute the Shapely value for each example-feature pair we can now use all fair baselines found with Alg. 2 at once.

4 Experiments

We evaluate in a classification task the local and global explanability power of our baseline method (**neutral**$_a$) against the **zero**, **average**, $\mathbf{p}_X$, and **mdb** baselines. Since the output of the trained classifier

Algorithm 2 Neutral Baseline Search for MLPs

Input: $G_\theta, a, Q, \delta, \epsilon$
Ouput: b

$queue \leftarrow []$
enqueue$([a], queue)$
for $l = L : -1 : 2$ **do**
  while queue is not empty **do**:
    $a \leftarrow$ dequeue$(queue)$
    for $j = 1 : |a|$ **do**
      $\text{SLP}_{(l,j)} = G^{(l)}(G_\theta^{(l-1)} \cdot \theta_j^{(l)})$ ▷ SLP Building
      $\tilde{a} \leftarrow$ **Algorithm 1**$(\text{SLP}_{(l,j)}, Q^{l-1}, a_j, \delta, \epsilon)$
      enqueue$(\tilde{a}, queue)$
    end for
  end while
end for
while queue is not empty **do**:
  $a \leftarrow$ dequeue$(queue)$
  $b \leftarrow b||a$
end while

is probabilistic, i.e., its image is $\in [0, 1]$ we set the neutrality value (a) to 0.5. This value simulates the case when the decision boundary is set to 0.5.

To do this we evaluate their global explanability power and local explanability power in two datasets: A less realistic dataset, a synthetic one, where we aim to compare the choice of baselines by limiting the drawbacks of the employed measures, and; A real dataset where we aim to validate these results in a more realistic scenario. In what follows, we first present the setup of evaluation methods used to quantify their explanability power. Then, we present the datasets and training setup. Finally, we present and discuss the results.

4.1 Evaluation Measures

To evaluate the global explanability power we use ROAR with average precision (i.e, area under the precision-recall curve) measured on the test set. We use average precision rather than accuracy because our datasets suffer from class unbalance. To evaluate the local explanability power we use a perturbation test similar to the one introduced by Ancona et al. (2017). Since we are evaluating probabilistic classifiers we do not measure deviations between predictions but a more appropriate measure of confidence, the *absolute logits*. Absolute logits, which are defined as $|\log(G_\theta(x)/(1 - G_\theta(x)))|$, where G_θ is a probabilistic classifier and x an example. These absolute logits can be used to measure prediction confidence, in fact when these increase it means that the confidence of the model in predicting one of the two classes is increasing and the opposite otherwise.

4.2 Datasets

We use two datasets, a synthetic and a real one. We use the former to simulate a dataset with independent features and controlled feature importance (the independence allow us to avoid the drawbacks of the evaluation methods above). The latter to experiment with a real use case.

Synthetic dataset. We generate a dataset with 5 independent features governed by the following sampling process. Given N, the number of examples x we want to generate, we sample each x from multivariate normal distribution with 0 mean and variance $I_{5\times5}$:

$$x \sim \mathcal{N}(0, I_{5\times5}) \tag{2}$$

This guaranties that features are independent with each other.

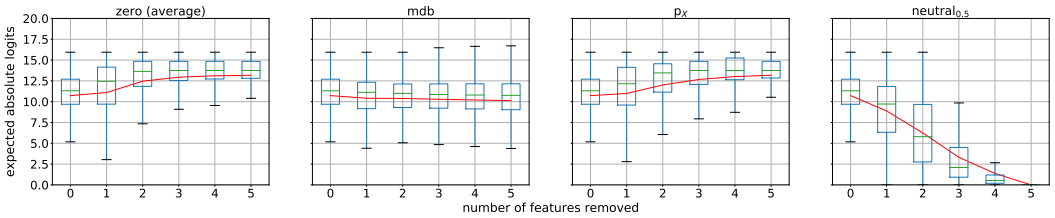


Figure 2: Local explanability power measured on the synthetic datasets (100 MC simulations).

We then define the importance of each feature by sampling a vector ϕ:

$$\phi_0 = \mu_0 \quad \text{with} \quad \mu_0 \sim \mathcal{U}(-15, 15) \tag{3}$$

$$\phi_j = (-1)^{\sigma} \cdot \mu \quad \text{with} \quad \sigma \sim \mathcal{B}ern(0.5),\ \mu \sim \mathcal{L}og\mathcal{N}(j, 1) \tag{4}$$

where ϕ_0 is sampled from a uniform distribution, and ϕ_j is decomposed into two values, a sign and a magnitude. The sign is sampled from a Bernoulli distribution with $p = 0.5$, while the magnitude is sampled from a log-normal distribution with mean j and variance 1. This makes, in expectation, the feature with the larger j more important than the one with a lower j.

We then add a constant term to each x to include a bias term and multiply this new vector to ϕ:

$$y^* = \phi \cdot ([1]||x)^\top \tag{5}$$

At this point we have generated x and y^* values. Since we are generating a synthetic dataset for a binary classification task, we define the label values (y), by transforming y^* as follows:

$$y = \begin{cases} 1 & \text{if } Q(\tau_1) < y^* < Q(\tau_1 + \tau_2) \\ 0 & \text{otherwise} \end{cases} \quad \text{with} \quad \tau_1, \tau_2 \sim \mathcal{U}[0.3, 0.5], \tag{6}$$

where Q is the quantile function of empirical CDF of y^*. This allows us to control the balance between positive and negative class, and make this dataset not solvable by a linear classifier.

Credit Card Default dataset. This is a dataset about default of credit card clients (Yeh and Lien, 2009). The dataset contains one binary target variable and 23 features. The number of observation are 29351. In order to be able to compute Shapely values on this dataset we reduced the number of features. To do this we applied PCA and extract the first two principal components within three sets of highly correlated features: pay, pay amt and bill amt; thus, we are left with a total of six transformed features and five plain.

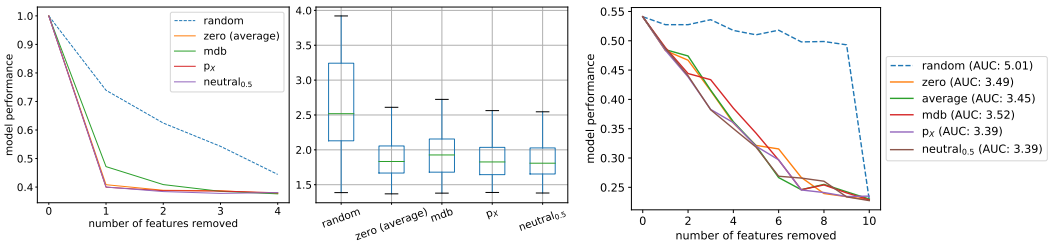
4.3 Experimental Setup

Before training the model, both datasets are preprocessed by standardising all non-categorical features. For the synthetic dataset we use a 2 layers MLP with activation functions $\tanh$ and sigmoid and 3 neurons in the hidden layer. While, for the CCD dataset, through cross-validation on the training set (80% of the dataset), we validated a 3 layers MLP with activation functions $\tanh$, $\tanh$, and sigmoid, with 10 and 6 neurons in the hidden layers. This model achieves an accuracy of 0.8070 on the test set (20% of the dataset). Both MLPs are trained using a binary cross entropy loss, and optimised using stochastic gradient descent.

The results obtained on the synthetic dataset are based on an 100 Monte Carlo (MC) simulations, where each simulation consists in generating a new synthetic dataset of 2000 examples and, on it, train an MLP and evaluate the attribution method with the various choice of baselines.

4.4 Results and Discussion

In Fig. 2 we show the result of the local evaluation of explanability power on the synthetic dataset. The box-plot and mean is computed over the 100 MC simulations where each time a new dataset is created. In the y-axis we see the expected absolute logits values, and in the x-axis, the number of features removed following the best feature detected by the attribution method with the various baselines. Note that because of the standardisation performed on the datasets, the zero and average



(a) Synthetic dataset (100 MC simulations). (b) CCD dataset.

Figure 3: Global explanability power evaluation (ROAR).

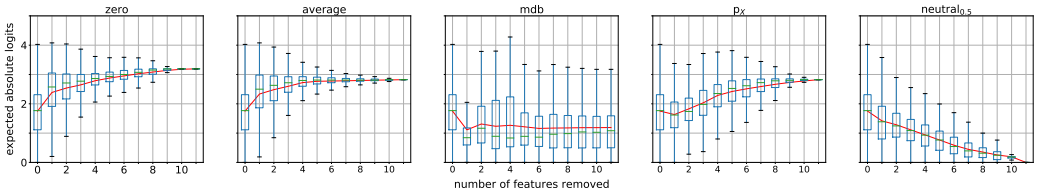


Figure 4: Local explanability power measured on the CCD dataset.

baselines are equivalent, therefore displayed once. Here, we observe that the only baseline that clearly decreases over the number of features removed is the $\text{neutral}_{0.5}$ baseline. This clearly suggests that this baseline is better in locally identifying the most important feature first.

In Fig. 3a we show the results for the global evaluation of explanability power on the synthetic dataset. Both plots are computed over the 100 MC simulations. The figure on the left shows the expected ROAR: in the y-axis we have the difference in performance measured when removing features, and in the x-axis we have the number of features removed as identified by the attribution method. The figure on the right is the expected box plots of the area under the curves of the figure on the left. For this last plot we perform a statistical test (a two-sided paired t-test, with $p < 0.05$): 1) all baselines are statistically significant when compared against the random baseline, and 2) the zero, p_X, and $\text{neutral}_{0.5}$ baselines are statistically significant against the mdb baseline. Here, we observe that when explaining the model globally, the choice of baseline does not affect the identification of the most important feature. All baselines with the exception of random and mdb are result to be equivalent. This makes sense since when evaluating the attribution of a feature globally, recalling our motivating example in Section 3, the majority of cases do not lie between the baseline and the decision boundary, which means that in expectation also a bad baseline is able to identify the most important features first.

In Fig. 4 we show an evaluation similar to the one in Fig. 2 but using the CCD dataset. Here, we observe that also in the real case, the neutral baseline is the only one that really decreases when removing features in order of discovery by the feature attribution method. This result confirms that also in the real case the proposed baseline outperforms the others.

In Fig. 3b we show the results for the global evaluation of explanability power on the CCD dataset. This plot is similar to the left plot of Fig. 3a. The AUC is here computed on the plotted curves. Here, we observe that the our and p_X baselines have the lowest AUC.

5 Conclusion

In this work, we have investigated the identification of baselines for Shapely value based attribution methods and MLPs. We have introduced the concept of neutrality and fair baselines. Their combination has allowed us, to develop a neutral baseline that empirically performs better than other choice of baselines. It remains open how to apply our method to recurrent networks and how to extend it to regression problems. Furthermore, inspired by the work of Chang et al. (2019), we plan to investigate the estimation of the joint CDFs via autoencoders in order to further optmise Alg. 1. A drawback of our methodology is that the computational cost of searching the baseline increases exponentially with respect to the number of hidden layers. We leave the analysis of possible solutions to these problems to future work.

Acknowledgments and Disclosure of Funding

We are grateful to the UCL Institute of Finance and Technology for financial sponsor. We thank Jan Hannes Lang for insightful inputs.

References

Ancona, M., C. Öztireli, and M. Gross, "Explaining Deep Neural Networks with a Polynomial Time Algorithm for Shapley Values Approximation," *36th International Conference on Machine Learning (ICML 2019)*, 2019.

Ancona, M, E Ceolini, C Öztireli, and M Gross, "A unified view of gradient-based attribution methods for deep neural networks," *NIPS 2017-Workshop on Interpreting, Explaining and Visualizing Deep Learning*, 2017.

Bach, Sebastian, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek, "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation," *PLOS ONE*, 07 2015, *10* (7), 1–46.

Caruana, Rich, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1721–1730.

Chang, Chun-Hao, Elliot Creager, Anna Goldenberg, and David Duvenaud, "Explaining Image Classifiers by Counterfactual Generation," in "7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019" OpenReview.net 2019.

Erion, Gabriel, Joseph D Janizek, Pascal Sturmfels, Scott Lundberg, and Su-In Lee, "Learning explainable models using attribution priors," *arXiv preprint arXiv:1906.10670*, 2019.

Goodman, Bryce and Seth Flaxman, "European Union regulations on algorithmic decision-making and a "right to explanation"," *AI Magazine*, 2017, *38* (3), 50–57.

Hooker, Sara, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim, "A benchmark for interpretability methods in deep neural networks," in "Advances in Neural Information Processing Systems" 2019, pp. 9734–9745.

Kapishnikov, Andrei, Tolga Bolukbasi, Fernanda Viégas, and Michael Terry, "XRAI: Better Attributions Through Regions," in "Proceedings of the IEEE International Conference on Computer Vision" 2019, pp. 4948–4957.

Kindermans, Pieter-Jan, Kristof T Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne, "Learning how to explain neural networks: Patternnet and patternattribution," *arXiv preprint arXiv:1705.05598*, 2017.

_ , Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim, "The (un) reliability of saliency methods," in "Explainable AI: Interpreting, Explaining and Visualizing Deep Learning," Springer, 2019, pp. 267–280.

Lundberg, Scott M and Su-In Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, 2017, pp. 4765–4774.

Montavon, Grégoire, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller, "Explaining nonlinear classification decisions with deep taylor decomposition," *Pattern Recognition*, 2017, *65*, 211–222.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

Samek, Wojciech, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller, "Evaluating the visualization of what a deep neural network has learned," *IEEE transactions on neural networks and learning systems*, 2016, *28* (11), 2660–2673.

Shrikumar, Avanti, Peyton Greenside, and Anshul Kundaje, "Learning important features through propagating activation differences," in "Proceedings of the 34th International Conference on Machine Learning-Volume 70" JMLR. org 2017, pp. 3145–3153.

Smilkov, Daniel, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg, "Smoothgrad: removing noise by adding noise," *arXiv preprint arXiv:1706.03825*, 2017.

Sturmfels, Pascal, Scott Lundberg, and Su-In Lee, "Visualizing the Impact of Feature Attribution Baselines," *Distill*, 2020, *5* (1), e22.

Sundararajan, Mukund and Amir Najmi, "The many Shapley values for model explanation," *arXiv preprint arXiv:1908.08474*, 2019.

_ , Ankur Taly, and Qiqi Yan, "Axiomatic attribution for deep networks," *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017, pp. 3319–3328.

Tsang, Michael, Dehua Cheng, and Yan Liu, "Detecting Statistical Interactions from Neural Network Weights," in "International Conference on Learning Representations" 2018.

Yeh, I-Cheng and Che hui Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," *Expert Systems with Applications*, 2009, *36* (2), 2473–2480.

Zeiler, Matthew and Rob Fergus, "Visualizing and understanding convolutional networks," in "European conference on computer vision" Springer 2014, pp. 818–833.

Supplementary Material for 'A Baseline for Shapely Values in MLPs: from Missingness to Neutrality'

Cosimo Izzo
University College London,
Now-casting Economics
cosimo.izzo.18@ucl.ac.uk

Aldo Lipani
University College London
aldo.lipani@ucl.ac.uk

Ramin Okhrati
University College London
r.okhrati@ucl.ac.uk

Francesca Medda
University College London
f.medda@ucl.ac.uk

1 Proof of Proposition 1

In this appendix, we provide the proof of Proposition 1 of our paper. For the convenience of the reader, some relevant assumptions and definitions of the paper are reviewed.

Assumptions:

A1. All activation functions are monotonic and continuous.
A2. All marginals cumulative distribution functions (CDFs) of the joint CDF of the input features are bijective and continuous.

Definition 1 (Neutrality Value)**.** *Given a model prediction $\hat{y}$ and a decision maker, we say that the value a is neutral if the decision maker's choice is determined by the value of $\hat{y}$ being either below or above a.*

Definition 2 (Space of Fair Baselines)**.** *Consider a dataset in $\mathbb{R}^k$, $k \geq 1$, generated by a distribution. A vector $x \in \mathbb{R}^k$ is in the space of fair baselines (hence called a fair baseline) if*

$$\max[C_j(x_j), 1 - C_j(x_j)] = \max[C_i(x_i), 1 - C_i(x_i)], \tag{1}$$

where C_is are the marginal CDFs, and the equality holds for any two C_i and C_j and the corresponding elements x_i and x_j of x.

Corollary 1. *Suppose that an SLP (G_θ) with the dataset of Definition 2 is given, and **A2** is satisfied. Then the set*

$$\tilde{B} = \{x^p \in \mathbb{R}^k : x_j^p = C_j^{-1}(\mathbb{1}_{\theta_j > 0} \cdot p + \mathbb{1}_{\theta_j < 0} \cdot (1-p)),\, p \in [0,1], j = 1, 2, \ldots, k\},$$

is a subset of the space of fair baselines.

Proof. Let us take an arbitrary $p \in [0,1]$ and assume that $x^p \in \tilde{B}$. The goal is to show that x^p is a fair baseline. We take two arbitrary elements of the vector x^p, let say x_i^p and x_j^p, $j \neq i$, and we show that (1) holds for each one of them.

Since $x_j^p \in \tilde{B}$ and C_j is bijective, we have that:

$$\begin{aligned} &\max[C_j(C_j^{-1}(\mathbb{1}_{\theta_j > 0} \cdot p + \mathbb{1}_{\theta_j < 0} \cdot (1-p))), 1 - C_j(C_j^{-1}(\mathbb{1}_{\theta_j > 0} \cdot p + \mathbb{1}_{\theta_j < 0} \cdot (1-p)))] \\ &= \max[\mathbb{1}_{\theta_j > 0} \cdot p + \mathbb{1}_{\theta_j < 0} \cdot (1-p), 1 - \mathbb{1}_{\theta_j > 0} \cdot p - \mathbb{1}_{\theta_j < 0} \cdot (1-p)] \end{aligned}, \tag{2}$$

a similar equality holds for x_i^p, i.e.

$$\begin{aligned} &\max[C_i(C_i^{-1}(\mathbb{1}_{\theta_i > 0} \cdot p + \mathbb{1}_{\theta_i < 0} \cdot (1-p))), 1 - C_i(C_i^{-1}(\mathbb{1}_{\theta_i > 0} \cdot p + \mathbb{1}_{\theta_i < 0} \cdot (1-p)))] \\ &= \max[\mathbb{1}_{\theta_i > 0} \cdot p + \mathbb{1}_{\theta_i < 0} \cdot (1-p), 1 - \mathbb{1}_{\theta_i > 0} \cdot p - \mathbb{1}_{\theta_i < 0} \cdot (1-p)] \end{aligned}. \tag{3}$$

Preprint. Under review.

The goal is to show that the right-hand sides of Equations (2) and (3) are indeed equal. For this to hold, we need to analyse four cases depending on the sign of θ_j and θ_i. We only provide the argument for the case $\theta_j > 0$ and $\theta_i < 0$, as the other three cases are fairly similar. Suppose that $\theta_j > 0$ and $\theta_i < 0$. Then the right-hand side of (2) becomes

$$\max[\mathbb{1}_{\theta_j>0} \cdot p + \mathbb{1}_{\theta_j<0} \cdot (1-p), 1 - \mathbb{1}_{\theta_j>0} \cdot p - \mathbb{1}_{\theta_j<0} \cdot (1-p)] = \max[p, 1-p],$$

and the right-hand side of (3) is equal to

$$\max[\mathbb{1}_{\theta_i>0} \cdot p + \mathbb{1}_{\theta_i<0} \cdot (1-p), 1 - \mathbb{1}_{\theta_i>0} \cdot p - \mathbb{1}_{\theta_i<0} \cdot (1-p)] = \max[1-p, p].$$

Obviously, these two equations are equal and so this proves the case of $\theta_j > 0$ and $\theta_i < 0$. □

Now, we are ready to prove the proposition.

Proposition 1 (Existence of a Fair Baseline for SLPs)**.** *Given an SLP (G_θ) satisfying **A1**, a dataset satisfying **A2**, and a neutrality value a in the image of G_θ, then there exists at least a fair baseline x such that $G_\theta(x) = a$.*

Proof. Since by Corollary 1, $\tilde{B}$ is a subset of the Space of Fair Baselines, it is enough to prove that $a \in G_\theta(\tilde{B})$ where $G_\theta(\tilde{B})$ is the image of $\tilde{B}$ under G_θ. Suppose that I is the image of the SLP. We show that $I \subseteq G_\theta(\tilde{B})$ which proves the result, since $a \in I$.

We start by showing that inf $G_\theta(\tilde{B}) \leq$ inf I and that sup $G_\theta(\tilde{B}) \geq$ sup I. Consider vector $x^0 \in \tilde{B}$ defined by $x^0 = \{x^0_j = C_j^{-1}(\mathbb{1}_{\theta_j<0}),$ for all $j = 1, 2, \ldots, k\}$. So elements of x^0 are the smallest possible if the coefficients are positive, and the largest possible when they are negative. From Assumption **A1**, it follows that $G_\theta(x^0)$ is the smallest value that the SLP can take. Hence, $G_\theta(x^0) \leq$ inf I.

Let us now take the vector $x^1 \in \tilde{B}$ which is defined by: $x^1 = \{x^1_j = C_j^{-1}(\mathbb{1}_{\theta_j>0}),$ for all $j = 1, 2, \ldots, k\}$. So elements of x^1 are the largest possible if the coefficients are positive, and the smallest possible when they are negative. From assumption **A1**, it follows that $G_\theta(x^1)$ is the largest value that the SLP can take. Hence, $G_\theta(x^1) \geq$ sup I.

Suppose that a is in the image of the SLP. Define function $h : [0, 1] \to G(\tilde{B})$ by $h(p) = G_\theta(x^p) = G(\theta \cdot (x^p)^\top)$ where $x^p \in \tilde{B}$, i.e, $x^p_j = C_j^{-1}(\mathbb{1}_{\theta_j>0} \cdot p + \mathbb{1}_{\theta_j<0} \cdot (1-p))$, $j = 1, 2, \ldots, k$. From the above argument, we have that $h(0) = G_\theta(x^0) \leq a \leq G_\theta(x^1) = h(1)$. Since G and C^{-1} are continuous functions by **A1** and **A2**, h is also continuous. By the intermediate value theorem, there is a $p^* \in [0, 1]$ such that $h(x^{p_*}) = a$, which means that $G_\theta(x^{p_*}) = a$. Finally, note that by **Corollary 1**, x^{p_*} is a fair baseline. This proves the proposition.

□