Counterfactual Shapley Additive Explanations

Emanuele Albini  
J.P. Morgan AI Research  
London, UK  
emanuele.albini@jpmorgan.com

Danial Dervovic  
J.P. Morgan AI Research  
London, UK  
daniel.dervovic@jpmorgan.com

Jason Long  
J.P. Morgan AI Research  
London, UK  
jason.x.long@jpmorgan.com

Daniele Magazzeni  
J.P. Morgan AI Research  
London, UK  
daniele.magazzeni@jpmorgan.com

1 INTRODUCTION

Government regulators are placing increasing emphasis on the fairness and discrimination issues in decision making processes using machine learning algorithms in high-stakes context as finance and healthcare. For example, U.S. credit regulations [50] put particular emphasis on the need to explain automatic decisions in terms of key factors that contributed to an adverse decision. Meanwhile, in the academic literature several techniques have been proposed to address this issue (see [1, 4, 10, 15] for an overview).

In the context of local explainability many approaches on which researchers have focused in the last years are based on the notion of feature attribution, i.e., distributing the output of the model for a specific input to its features (e.g., LIME [40], SHAP [27], GIG [29]). In this paper in particular we will focus on SHAP, one of the most popular techniques to generate local explanations based on the notion of Shapley value [42] from game theory. Shapley value-based frameworks for Explainable AI (XAI) consider each feature as a player in a $m$-person game to fairly distribute the contribution of each feature to the output of the model. To do so they compare the output of the (same) model when a feature is present with that of when the same feature is missing. There are two main limitations with this approach that have been raised in the literature:

(a) It is not clear how to define the output of the model when a feature is missing. The most common approach is to estimate it as an expectation over a background distribution of the input features [28].

(b) There is no explicit guidance provided on how a user might alter one's behavior in a desirable way [23].

Another popular area of research has developed around counterfactual explanations, also known as algorithmic recourse, i.e., given a specific input one must find the "closest possible world (input)" [53] that gives rise to a different outcome. In practise, this means that these approaches aim to find one (or more) points that are (1) close to the one we want to explain; and (2) "plausible" (where plausibility can be defined in different ways in the literature, see [22] for more insights). Counterfactual explanations have two main limitations:

(a) Most of the approaches in the literature are limited at finding a single counterfactual point. While this may give the user a clear understanding of what they could do in order to reverse an adverse outcome, it does not allow them to choose changes that are more suited for them.
(b) While there has been some attempt at generating diverse sets of counterfactual points (e.g., [32, 41]), there is no consensus on how to limit the cognitive load for the user caused by the sheer amount of information that is provided, or – in other words – on how to provide a more amenable explanation (in terms of size), as advocated also from a social science perspective [30].

In this paper we present how these two general approaches for explainability can be combined in order to provide a counterfactual feature attribution grounded on the game-theoretic approach afforded by Shapley values that we call Counterfactual SHAP (CoSHAP). We are motivated by the desire to retain the simple form of explanation provided by feature attributions, while introducing the actionability properties of counterfactual explanations. In particular, our contributions are as follows.

- We enumerate the assumptions that are necessary to interpret Shapley values in a counterfactual sense and discuss what it means for a feature attribution method to demonstrate counterfactual behaviour.
- We introduce the notion of counterfactual-ability of a feature attribution as a way to quantitatively evaluate its ability to suggest to the user how to act upon the input in order to overcome an adverse prediction.
- We propose to use (a uniform distribution over) a set of counterfactual points as the background distribution for the computation of Shapley values in order to achieve higher counterfactual-ability, yielding the CoSHAP algorithm.
- We benchmark the CoSHAP algorithm using a number of different counterfactual generation techniques from the literature against baseline feature attribution techniques. CoSHAP (using 10-NN∗ as the counterfactual generation technique) is shown to have the best counterfactual-ability on several public datasets taken from the financial domain.

We note that in this paper we concentrate on tree-based models for the following reasons: (1) in the context of classification and regression for tabular data, tree-based ensemble models as XGBoost, CatBoost and LightGBM are deemed as the state-of-the-art in terms of performance and therefore are widely adopted in many industries including finance [48]; (2) interventional Shapley values can be computed exactly for tree-based models using the algorithm proposed in [26].

We also note that in this paper we will put particular emphasis on models that are monotone, i.e., models in which their output is (positively or negatively) monotonic with respect to each of the input features. We do this for two reasons: (1) monotonicity is a key precondition for the interpretation of a feature attribution in counterfactual terms, as will be described in detail in Section 3.2; (2) high stakes models (that are usually the ones in most need of explanations) are usually constrained to be simpler at design-stage with the objective of making them more interpretable. One such common constraint is imposing some kind of monotonic relationship between the features and their output [48].

2 BACKGROUND

In the remainder of this paper we consider a binary classification model \( f : X \to Y \) where \( X = \mathbb{R}^m \) and \( Y = \mathbb{R} \). We define the decision function \( F : X \to \{0, 1\} \) as follows:

\[
F(x) = \begin{cases} 
1 & \text{if } f(x) > 0 \\
0 & \text{otherwise}
\end{cases}
\]

We refer to \( f(x) \) as the model output and to \( F(x) \) as the model prediction or outcome. Note that, as reported in the definition of \( F(x) \) and without loss of generality, we use 0 as decision threshold for the binary prediction. Moreover, without loss of generality, in the sequel we assume that an input \( x \in X \) such that \( F(x) = 1 \) is an adverse outcome for the user. We also note that all the results in this paper can be trivially generalized to multi-class models.

2.1 Shapley values

The Shapley values method is a technique used in classic game theory to fairly attribute the payoff to the players in an \( m \)-player cooperative game. Formally, given a set of players \( \mathcal{F} = \{1, \ldots, m\} \) and the characteristic function of the game \( v : 2^{\mathcal{F}} \to \mathbb{R} \), the Shapley value of player \( i \) is defined as:

\[
\phi_i = \sum_{S \subseteq \mathcal{F} \setminus \{i\}} \left( \begin{array}{c} m-1 \\ |S| \end{array} \right)^{-1} [v(S \cup \{i\}) - v(S)]
\]

In the context of machine learning models the players are the features of the model and several ways have been proposed to simulate feature absence in the characteristic function (e.g., retraining the model without such feature [47], or using the conditional or marginal expectations over a background distribution [27]).

In this paper we use the approximation of the characteristic function proposed in [27] and [26] that simulates the absence of a feature using the marginal expectation over a background distribution \( \mathcal{D} \):

\[
v(S) = \mathbb{E}_{x \sim \mathcal{D}} \left[ f(x_S, x_{\mathcal{F} \setminus S}) \right]
\]

where with an abuse of notation \( f(x_S, x_{\mathcal{F} \setminus S}) \) indicates the output of the model with feature values \( x \) for features in \( S \) and values \( x' \) for feature values not in \( S \). In the remainder of this paper we will refer to the space of Shapley values \( \mathbb{R}^m \) as \( \Phi \) and the Shapley values vector of \( x \) as \( \phi \).

2.2 Counterfactual Explanations

In its basic form, a (local) counterfactual explanation (CF) for an input \( x \) is a point \( x' \) such that (1) \( x' \) gives rise to a different prediction, i.e., \( F(x) \neq F(x') \), (2) \( x \) and \( x' \) are close (under some distance metric) and (3) \( x' \) is a “plausible” input. This last constraint has been interpreted in several ways in the literature, it may involve considerations about sparsity (e.g., [44]), closeness to the data manifold (e.g., [34]), causality (e.g., [21]), actionability (e.g., [36, 51]) or a combination thereof (e.g., [11]). A plethora of techniques for the generation of counterfactuals exist in the literature using search algorithms (e.g., [2, 45, 53]), optimization (e.g., [19]) and genetic algorithms (e.g., [43]) among other methods (we refer the reader to [20, 22, 46, 52] for recent surveys).

\[\text{We use lower-case bold symbols to indicate vectors.}\]
3 INTERPRETING SHAPLEY VALUES IN COUNTERFACTUAL TERMS

In general, Shapley values do not have an obvious interpretation in counterfactual terms, this means that they do not provide suggestions on how a user can change their features in order to change the prediction [3, 23]. We argue that this is due to 2 main reasons: (1) the “arbitrary” choice of background distribution for the computation of Shapley values and (2) the lack of monotonicity constraint of the model. We now discuss the details of these two reasons.

3.1 Choice of the background distribution

Shapley values describe the contributions of the players (features) to the game payoff (model output). In the context of machine learning model explainability an important assumption is made: the simulation of each feature’s absence in the cooperative game using a background distribution $D$. As pointed out in [28], this means that Shapley values explain a prediction of an input in contrast to a distribution of background points. In practise, the background distribution is taken as a uniform distribution over unit point masses at a finite number of points, called the background dataset.

Therefore, the background dataset should be chosen according to the contrastive question that we aim to answer. We list some of the most common distributions that have been proposed/used.

- Training Set ($D_{\text{TRAIN}}$) [27]. The whole training set, including the samples that are labelled and/or predicted of being of the same class of the input.
- Differently-Labelled Samples ($D_{\text{DIFF-LAB}}$). The samples in the training set labelled differently than the input.
- Differently-Predicted Samples ($D_{\text{DIFF-PRED}}$). The samples in the training set predicted with a different class.
- Differently-Predicted Samples Median ($D_{\text{DIFF-MEDIAN}}$). A single point obtained as the feature-wise median of the points predicted with a different class.

These choices of background dataset have in common the fact that they are defined a priori, i.e., they are equal for all the inputs. This means that we are contrasting an input $x$ with a (input-invariant) distribution $D$ that may potentially be very different from $x$. This can give rise to explanations that are sometimes misleading for a user who is typically interested in understanding which features led to the adverse outcome (in order to reverse it). In other words the constrastive question that we are answering with the Shapley values is not tailored to the specific input (user) and therefore instead of answering the question of “Why was a user rejected when compared to similar users that were accepted?” we will be answering the question of “Which features are most important in making my outcome different from that of other (accepted) users?” (potentially very different from $x$).

If we consider the example in Figure 1.i that shows an explanation where background dataset is the training set, we note that the Shapley values suggest that Feature $A$ negatively contributed to the model output; this means that the current value of Feature $A$ is “protective” against rejection when put in contrast with the expected output of the model obtained when using the background distribution $D_{\text{TRAIN}}$. This may be useful information for the model developers but it does not allow one to gain any (actionable) insight unless we assume access to the underlying distribution $D_{\text{TRAIN}}$. In fact, this explanation only informs the user that its Feature $A$ is better than the one of an average customer but it does not either (a) advise them on how they can change their features in order to overcome the (adverse) outcome; or (b) inform them on which features were most important in rejecting their application.

Figures 1.ii, 1.iii and 1.iv show how alternative (but still input-invariant) background distributions ($D_{\text{DIFF-LAB}}, D_{\text{DIFF-PRED}}$ and $D_{\text{MEDIAN}}$, respectively) may improve the explanations in terms of informing the user on which features were most important in rejecting their application when compared to other rejected samples, but they still lack the ability of giving useful insight on which features were the most important and therefore should be acted upon in order to reverse the adverse decision $F(x) = 1$. This is due to the contrastive question being posed with respect to (a) samples that have much better (lower) model outputs and (b) samples that have similar model output but that are very different from $x$. 

Figure 1: Effect of different choices of background dataset on the Shapley values ($\phi$) of the same input ($x$) with the same model. Red regions correspond to areas of the feature space where the decision is adverse, i.e. $F(x) = 1$, with blue regions representing the opposite, i.e. those $x \in X$ for which $F(x) = 0$. Coloured arrows and scatter points represent the directions of the Shapley values vector and the background dataset used for their computation, respectively.
who is given an explanation that contrasts them with the average

As remarked in [3], feature attributions do not clearly provide guidance on how to alter the features in order to change the prediction of a model, and sometimes they can be even misleading in that respect because they make the assumption that the model is monotone.

Figure 2: Effect of the lack of monotonicity constraint on the Shapley values of two same inputs using the 10-NN as background dataset. White arrows indicate the monotonic trends of the model. Coloured arrows and scatter points represent the directions of the Shapley values vector and the background dataset used for their computation, respectively.

Using a set of counterfactual points as the background dataset solves the issues mentioned in the preceding example. In particular, using counterfactuals as the background dataset allows one to answer a contrastive question that is (a) of interest for the user because it is comparing $x$ to samples that are similar to them (and implicitly more "reachable") and (b) more amenable in terms of access to the underlying distributions. In fact, as mentioned earlier, a useful interpretation of Shapley values-based explanations requires access to the background distribution. Arguably, a user can relate to a set of similar customers more easily than the training set (that may contain very different users). For example, imagine a fraud-prone millionaire being rejected for a consumer trading account who is given an explanation that contrasts them with the average customer (who is very likely neither fraud-prone nor a millionaire).

For example, if we consider Figure 1.v that shows the direction of the Shapley values calculated using the 10 nearest neighbors of the input that were accepted, we note how both features are deemed as contributing to the rejection when compared to similar customers that were instead accepted. And in fact, Feature A has a higher importance than Feature B since the model is locally more sensitive to Feature A than Feature B as shown by the sharper color gradient in the horizontal direction.

As noted earlier, using a set of diverse counterfactual points as a background distribution (contrary to "classic" input-invariant background distributions) means that the background distribution depends on the input $x$. Therefore, in the sequel we will denote such distributions as $D_C(x)$ where $C$ is the name of the counterfactual technique used to generate the background dataset. For simplicity of exposition, from now on we refer to explanations using a diverse set of counterfactual points as the background distribution for the computation of Shapley values as Counterfactual SHAP or, for short, CoSHAP.

3.2 Monotonicity of the model

As remarked in [3], feature attributions do not clearly provide guidance on how to alter the features in order to change the prediction of a model, and sometimes they can be even misleading in that respect because they make the assumption that the model is monotone.

For instance, if we consider Figure 2.i showing the Shapley values of two points, $x^{(0)}$ and $x^{(1)}$, for a monotonic model, we note how both explanations have a positive Shapley value for Feature A but in order to overcome the adverse outcome Feature A must be increased for $x^{(0)}$ while it must be decreased for $x^{(1)}$. This means that a user (that has no access to the model) is unable to move the feature in the most sensible direction. In contrast, we observe in Figure 2.ii – where the model is monotone in both features A and B – that the Shapley values for $x^{(0)}$ and $x^{(1)}$ both again have similar Shapley values. Changing Feature A in the same direction as the sign of its Shapley value will decrease the value of the model output for both of the points $x^{(0)}$ and $x^{(1)}$, thereby yielding a better outcome for both points. In fact, if the model is monotone then moving a feature with a positive (negative) monotone trend in one direction will give rise to a predictable change in the model output in the same (opposite) direction.

Having described how the background distribution used for the computation of Shapley values and the monotonic behaviour of the model play a key role in giving a counterfactual interpretation to Shapley values, we now turn to the open question on how we can numerically measure this "counterfactual-ability" of a feature attribution. We will tackle this problem in Section 4.

4 COUNTERFACTUAL-ABILITY

We seek to formalise the notion that certain feature attributions will be more useful for a model user in changing features to reverse an adverse outcome. It is important to emphasise that predicting how users might engage with explanations is a very challenging problem, and behaviour may vary dramatically depending on the context. We do not claim to resolve this problem. However, we aim to set up a flexible framework to measure the ability of an explanation to help a user reverse an adverse decision, before specialising this framework under certain plausible assumptions about how a user could act on the explanations that they receive.

Counterfactual-ability. To define the counterfactual-ability of a feature attribution $\phi$ we will measure the cost that a user will incur when changing the input $x$ into a new input $x'$ based on the information provided by the feature attribution $\phi$.

We will measure the cost of changing an input $x$ into another input $x'$ via a cost function. Formally, a cost function is a function $c : X \times X \rightarrow \mathbb{R}^+$ where $c(x, x')$ is the cost for the user of moving from $x$ to $x'$. A very simple example of cost function could be the Euclidean distance defined in the input space.

In order to describe how a user acts upon the input $x$ based on the information provided by the feature attribution $\phi$ we use an action function. Formally, an action function is a function $A : X \times \Phi \rightarrow 2^X$ where $A(x, \phi) \subseteq X$ is a subset of the input space describing plausible changes the user may enact upon $x$ when provided with an explanation $\phi$. We will refer to $A(x, \phi)$ as the action subset. Note that we do not constraint the action subset to be finite.

Intuitively the output of an action function can be interpreted as a subset of the possible options that a user may consider when changing the input based on the information provided by the feature attribution. For instance, a user may consider as possible options only changes to the most important feature according to the feature attribution. In the most extreme scenario a user may ignore the
Choice of action function. After defining the general concepts of action and cost functions we now define a concrete instance that we use in this paper. To do so, we start with a number of assumptions. These assumptions are designed to create a sensible metric for the counterfactual-ability of an explanation in the context of algorithmic recourse. Intuitively, the assumptions aim to cast the feature attribution as a suggested direction for a user to move in feature space, and the counterfactual-ability of the a feature will therefore measure the distance (under a sensible metric) to the decision boundary along this line.

**Action Function.** In the scope of this paper, we use the following set of assumptions:

- **Monotonic recourse.** When changing a feature a user will move its value in the opposite direction of the monotonic trend, e.g., to reduce the risk of default a user will try to increase their income (as opposed to reducing it).
- **Adverse factors recourse.** A user will change the features with positive Shapley values, i.e., the features contributing to the adverse prediction (as opposed to also improving features that are already good).
- **Proportional recourse.** A user will change the features with the highest (positive) Shapley values changing them proportionally to their Shapley values.
- **Recourse cost.** When moving feature proportionally to their Shapley values we use the quantile shift as metric to compare the cost of the recourse.

We will call the action function satisfying these assumptions $\hat{A}_k$ where $k$ is the number of top features that a user is considering. We formally define it as follows
\footnote{We use $\cdot$ and $\odot$ to indicate the dot and element-wise product, respectively.}

$$\hat{A}_k(x, \phi) = \{x' : Q(x') - Q(x) = -\lambda \phi \odot T \odot \mathbb{I}_k(\phi), \forall \lambda > 0, x' \in X\}$$

where $T \in \{-1, +1\}^m$ is the trend vector, satisfying $T_k = 1$ if the trend of feature $k$ is monotonically positive and $T_k = -1$ if the trend is monotonically negative; $Q: X \rightarrow [0, 1]^m$ is a function computing the quantile (marginal cumulative distribution function) of each of the features with respect to the distribution induced by the training data; and $\mathbb{I}_k(\phi) \in \{0, 1\}^m$ is a (binary) indicator vector for the top-$k$ features in $\phi$. Formally:

$$\mathbb{I}_k(\phi) = \begin{cases} 1 & \text{if } i \in \arg \max_{S \subseteq \{1, \ldots, m\}, |S| \leq k} \sum_{i \in S} \phi_i \\ 0 & \text{otherwise} \end{cases}$$

The intuition behind this choice of action function is that the feature attribution should provide a suggested direction to the user that takes them towards the decision boundary. However, realistic actions will not involve changes to every feature; rather, a user may focus on making changes to only the top-$k$ most important features, and we reflect this in our choice of action function. We use the quantile shift as a normalised metric for recourse cost, so that the action subset induced by our action function is a semi-infinite line in the normalized quantile input space in the direction of the Shapley vector with its sign adjusted to match the monotonic trend. To better understand this concept we can consider Figure 3,
showing an example of the action subset induced for an input \( x \) and an attribution \( \phi \).

We note that our choice of action function is just one instantiation of the framework that we propose. We argue that casting the explanation as a direction in which an input point may move is a natural choice that allows for concrete comparisons between methods, but we acknowledge that there is no clear answer to the question of how different users may act upon \( x \) given \( \phi \) in full generality. We believe that this topic represents an interesting future research direction, and we discuss this further in Section 7.

Cost Function. We measured the cost using the quantile shift under L1 and L2-norm, common metrics in the actionable recourse literature [51]. Formally:

\[
c_{L1}(x, x') = \| Q(x') - Q(x) \|_1, \quad c_{L2}(x, x') = \| Q(x') - Q(x) \|_2.
\]

### 5 EXPERIMENTS

In order to understand how different variants of CoSHAP perform in terms of counterfactual-ability we compared them against existing feature attribution techniques. Table 1 describes in detail the feature attributions that we considered in our experiments.

In particular, we considered 3 variants of Counterfactual SHAP that differs from each others for the technique used to generate counterfactual points that are as follows.

- Feature Tweaking (FT) [49]. We take all the \( \epsilon \)-perturbations of the input that lead to a different prediction. In our experiments we used \( \epsilon = 10^{-6} \).

- \( K \)-Nearest Neighbours (K-NN) [33]. We take the \( K \) nearest points to \( x \) in the training set, referred to as \( D_{KNN}^x \), such that their predictions are different from \( x \)‘s. In our experiments we used \( K = 10 \) and the euclidean distance over the quantile space as distance metric.

- Decision Boundary \( K \)-Nearest Neighbours (K-NN\(^*\)). Since the counterfactual points generated using K-NN are samples extracted from the training set they tend to be at a greater distance from the decision boundary (DB) than artificially generated counterfactual points (for instance those generated using FT). For this reason we generated a variant of K-NN obtained by intersecting the DB with the lines connecting \( x \) with the \( K \) nearest points\(^3\).

For comparison with CoSHAP, we considered baselines belonging to two broad families of feature attribution methods from the literature. On the one hand, we compared CoSHAP with Shapley values obtained using common input-invariant background distributions: \( D_{TRAIN} \), \( D_{DIFF-PRED} \), \( D_{DIFF-LAB} \) and \( D_{DIFF-MEDIAN} \). We refer to Section 3.1 for more details about these distributions. On the other hand, we compared CoSHAP with existing feature attribution techniques (non-Shapley values-based) that have a counterfactual intent. In particular we considered the frequency-based approach proposed in [31], that for each feature generates the attribution score as the fraction of counterfactual points possessing a modified value of the feature with respect to the input \( x \). In order to generate a (diverse) set of counterfactual points we used the same techniques that we used to generate the background datasets for CoSHAP described earlier in this section, i.e., FT, K-NN, K-NN\(^*\).

### Setup.

To run the experiments we used 4 publicly available datasets: GMSC (Give Me Some Credit) [17], HELOC (Home Equity Line Of Credit) [13], LC (Lending Club Loan Data) [18] and WINE (UCI Wine Quality) [9]. For each dataset we used a 70/30 split. We trained a monotonic XGBoost model [8], using the Spearman’s Rho (with respect to the target variable) to determine the monotonic trend of the features. We hyper-trained the parameters using Bayesian optimization via hyperopt [5] for 1000 iterations maximizing the average ROC-AUC under a 5-fold cross validation.

As described in Section 2, we used 0 as decision threshold in the raw score space (log odds) for the binary prediction.

### Results.

We measured the percentage of times in which CoSHAP performs better in terms of counterfactual-ability than baselines over 1000 rejected (i.e. with \( f(x) = 1 \)) random samples close to the decision boundary (for which \( \sigma(f(x)) \leq 0.6 \)) for each of the 4 datasets. Figure 4 and Figure 5 show the results using the cost functions \( c_{L1} \) and \( c_{L2} \), respectively. We report the main findings.

- CoSHAP 10-NN and CoSHAP 10-NN\(^*\) consistently beat (i.e., > 50%) all of the baselines, performing between 51.2% and 96.4% of the cases better than the baselines. We note that further investigation is necessary to fully understand the effects of the hyper-parameters of K-NN\(^*\) on the resulting CoSHAP explanation and how they related with the size and dimensionality of the training data.

- For several datasets, CoSHAP FT does not perform well against the baselines (i.e., in more than 50% of the considered samples the counterfactual-ability is lower than that of the baselines). In particular, CoSHAP FT fails to beat the performance of the (baseline) SHAP DIFF-PRED that uses

\(^3\)\( X_{KNN} = \{x_{KNN}^*: x_{KNN}^* = (x_{KNN}^* \cdot \chi(x_{KNN}^*)) \cap DB, \forall x_{KNN}^* \in X_{KNN}\} \) where \( x_{KNN}^* \cdot \chi(x_{KNN}^*) \) denotes the line segment between \( x \) and \( x_{KNN}^* \) and \( X_{KNN} \) and \( X_{KNN}^* \) are the CoSHAP K-NN and CoSHAP K-NN\(^*\) background datasets, respectively. In practice, we obtained \( X_{KNN} \) by simply applying the bisection method in order to find the intersections.

\(^*\)\( \sigma \) denotes the sigmoid function.

| Method Name | Type\(^1\) | Distribution\(^2\) |
|-------------|-------------|------------------|
| CoSHAP FT   | Shapley     | \( D_{FT}(x) \)  |
| CoSHAP 10-NN| Shapley     | \( D_{10-NN}(x) \) |
| CoSHAP 10-NN\(^*\) | Shapley | \( D_{10-NN}(x) \) |

Table 1: Explanation methods used in the experiments divided among Counterfactual SHAP variants and baselines. (*) Variant of the distribution where points are projected on the decision boundary (see Section 5); (\( \dagger \)) type of feature attribution, i.e., Shapley values or Frequency-based feature attribution; (\( \ddagger \)) distribution used as background (for Shapley values) or to generate a (diverse) set of counterfactual points (for frequency-based feature attribution), refer to Section 3.1 for details.
Figure 4: Percentage of times in which the counterfactual-ability $CA(x, \phi, \tilde{A}_k, c_L)$ of different versions of CoSHAP (CoSHAP $10$-NN, CoSHAP $10$-NN$^*$ and CoSHAP FT) is higher (better) than the counterfactual-ability of the baseline feature attributions (SHAP TRAIN, SHAP DIFF-LAB, SHAP DIFF-PRED, SHAP DIFF-MEDIAN, FREQ $10$-NN and FREQ FT). The plots show how the counterfactual-ability changes when varying the number of top-$k$ features considered in the action function. Each line represents a baseline.

- The full set of samples predicted as belonging to a different class as background dataset. This is due to the sparsity of the counterfactual points generated by FT.
- The results are robust with respect to the norm (L1 or L2) used to aggregate the cost of different features.

6 RELATED WORK

There has been recently an increasing interest in exploring the relationship between feature importance and counterfactual explanations.

A recent work [54] has proposed a Bayesian decision theory-based approach to the computation of the Shapley values. In particular the idea of [54] is to optimize the choice of the background distribution for the computation of Shapley values maximizing the expected reward for the user, i.e., $D^* = \arg\max_{D' \subseteq D} \mathbb{E}_{x \sim D'} [r(x)]$, under a certain reward function $r$. The work provides a theoretical framework for modelling user preference and beliefs but lacks (by design) concrete (1) guidance on how to select $D^*$, (2) how to update the reward function $r$ based on the observed Shapley values and (3) how to interpret the feature attribution $\phi$ into practical actions on the input $x$ in order to (automatically) solve the optimisation problem without resorting to an update in human-in-the-loop fashion.

Other works have proposed to fill the gap between feature attribution techniques and counterfactual explanations by different means than Shapley values. In particular, [31] and [43] propose techniques to generate feature attributions from a set of diverse counterfactual points but (contrary to us) they use frequency-based approaches, i.e., they give higher attribution to features that are more often changed in counterfactual points. This implies that also features potentially ignored by the model may receive a high feature importance because they are correlated with other features that are really used by the model. As remarked in [7] this behaviour may be desirable in some context as medical sciences but not in others, as in the credit scoring scenario in which users are ultimately interested in understanding why they have have been rejected by the model rather than which features correlate with rejection in the data. We used [31] as a baseline in the experiments in Section 5. In [6] feature attributions are generated by approximating the minimal adversarial perturbation using an adversarially trained neural network on a (differentiable) neural network-based surrogate model. This approach tends to follow the most strictest interpretation of the “true to the model” paradigm [7] enforcing only the class change but does not directly allow for the enforcement of other constraints, e.g., regarding the plausibility of such changes, as we do by providing a background distribution that is based on counterfactuals.

Other works such as [12, 37, 38] analyze the complementary problem to that we analyze in this paper: they show how feature attributions can guide the search of counterfactual points (while we investigate how different techniques for the generation of counterfactual examples can empower better feature attribution).

In general, many works have explored how to evaluate counterfactual explanations (e.g., [25, 39, 51]) and feature attributions (e.g., [14, 24, 35]) but few proposed a quantitative metric to evaluate feature attributions in counterfactual terms. In [55] the authors propose to evaluate feature attributions with a fidelity error for each of the features that (differently from counterfactual-ability) is computed changing only a single feature at a time. We also note that in this paper, our definition of counterfactual-ability is inspired by the notion of quantile shift cost proposed in [51], originally designed to evaluate counterfactual explanations (while counterfactual-ability is a metric for feature attributions).
Towards the more general goal of unifying feature attribution techniques and counterfactual explanations, we have shown how using counterfactual points as the background distribution for the computation of Shapley values (CoSHAP) allows one to obtain feature attributions that can better advise towards useful changes of the input in order to overcome an adverse outcome. We proposed a new quantitative framework to evaluate such an effect that we called counterfactual-ability, and remarked on the role that monotonicity of the model plays in the generation of feature attributions with a counterfactual intent. We evaluated CoSHAP on 4 publicly available datasets and highlighted that using simpler counterfactual techniques such as those based on nearest-neighbours within CoSHAP performs better than existing feature attribution methods.

Our proposal can be extended in several directions. Firstly, it would be interesting to explore alternative notions of action and cost function, grounding their definition with findings in psychology and the social sciences concerning how users interpret feature attributions and how they consequently change their behaviour. For example, one possibility would be to expand the definition of action function to take into account user preferences for certain actions – this could be achieved by coupling each “possible action” returned by the action function with a probability. Secondly, investigating additional metrics for the evaluation of a feature attribution in counterfactual terms would also be of interest. For instance, one could include considerations about the sparsity and/or the plausibility of the feature attributions, as well as other metrics drawn from the counterfactual explanations literature. Lastly, testing our approach on different models (e.g., neural networks) and using a wider variety of (potentially model-agnostic) counterfactual explanation techniques as [11, 16, 19–21, 39] represents another interesting future direction.

From a wider perspective, our work draws attention to some gaps in the literature that we believe are worthy of further investigation. On the one hand, the importance of techniques for the generation of a diverse set of counterfactuals advocated by many practitioners [32, 41, 44]. On the other hand, it highlights how few techniques have the capabilities of generating diverse counterfactual explanations in the context of non-differentiable models, such as ensembles of decision trees that are among the most widely adopted in industry.

ACKNOWLEDGMENTS

Disclaimer. This paper was prepared for informational purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co. and its affiliates (“JP Morgan”), and is not a product of the Research Department of JP Morgan. JP Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

REFERENCES

[1] Amina Adadi and Mohammed Berrada. 2018. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). IEEE Access 6 (9 2018), 52138–52160.
[2] Emanuele Albini, Antonio Rago, Pietro Baroni, and Francesca Toni. 2020. Relation-Based Counterfactual Explanations for Bayesian Network Classifiers. In Proc. of the 29th Int. J. Conf. on Artificial Intell., IJCAI 451–457.
[3] Solon Barocas, Andrew D Selbst, and Manish Raghavan. 2020. The Hidden Assumptions Behind Counterfactual Explanations and Principal Reasons. In Proc. of the 2020 Conf. on Fairness, Accountability, and Transparency, FAccT 86—89.
Counterfactual Shapley Additive Explanations

[4] Alejandro Barredo Arrieta, Natalia Diaz-Rodriguez, Javier Del Ser, Adrien Benetot, Siham Tabik, Alberto Barbadó, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chattia, and Francisco Herrera. 2020. Explainable and Fairly Accurate Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion 58 (2020), 82–115.

[5] James Bergstra, Daniel Yamins, and David Cox. 2013. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In Proc. of the 30th Int. Conf. on Int. Conf. on Machine Learning, ICML 1–115–123.

[6] Matt Chapman-Rounds, Umnang Bhatt, Erik Pazo, Marc-André Schulz, and Konstantinos Georgatidis. 2021. TINAF: Feature Importance by Minimal Adversarial Perturbation. In Proc. of the 55th AAAI Conf. on Artificial Intell., AAAI 15, 35 (5 2021), 11433–11441.

[7] Hugh Chen, Joseph D Janzick, Scott Lundberg, and Su-In Lee. 2020. True to the Model or True to the Data?. In IJCM ’20 Workshop on Human Interpretability. arXiv:2006.16234

[8] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD ’16: 785–794.

[9] Paulo Cortez, António Cereira, Fernando Almeida, Telmo Matos, and José Reis. 2009. Modeling wine preferences by data mining from physicochemical properties. Decision Support Syst. 47, 4 (11 2009), 547–553.

[10] Kristijonas Čyras, Antonio Rago, Emanuele Albini, Pietro Baroni, and Francesca Toni. 2021. Argumentative XAI: A Survey. In Proc. of the 29th Int. J. Conf. on Artificial Intell., IJCAI Vol. 5. 4392–4399.

[11] Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. 2020. Multi-Perspective Counterfactual Explanations. In Proc. of the 16th Int. Conf. on Parallel Problem Solving from Nature, Vol. 12269 LNCs. 448–469.

[12] Carlos Fernández-Loria, Foster Provost, and Xintian Han. 2021. Explaining Data-Driven Decisions Explaining Data-Driven Decisions made by AI Systems: The Counterfactual Approach. arXiv:2001.07417.

[13] FICO Community. 2019. Explainable Machine Learning challenge. https://community.fico.com/s/explainable-machine-learning-challenge

[14] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, and Fosca Giannotti. 2018. Rule-Based Explanations of Black Box Decision Systems. arXiv:1805.10820

[15] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Fosca Giannotti, and Dino Pedreschi. 2019. A Survey of Methods for Explaining Black Box Models. Comput. Surveys 51, 5 (2019), 1–42.

[16] Masoud Hashemi and Ali Fathi. 2020. PermuteAttack: Counterfactual Explanation of Machine Learning Credit Scorecards. arXiv:2008.10138

[17] Kaggle. 2019. Give Me Some Credit. https://www.kaggle.com/c/GiveMeSomeCredit

[18] Kaggle. 2019. Lending Club Loan Data. https://www.kaggle.com/wordsforthewise/lending-club

[19] Kentaro Kanamori, Takuya Takagi, Ken Kobayashi, and Hiroki Arimura. 2020. A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data. SEDIC, LIME-C and SHAP-C. Adv. in Data Analysis and Classification 14 (2020), 801–819.

[20] Shubham Rathi. 2019. Generating Counterfactual and Contrastive Explanations using SHAP. In 2nd Workshop on Humanizing AI (HAI) at IJCAI ’19. arXiv:1906.09293

[21] Kaivalya Rawal and Himabindu Lakkaraju. 2020. Beyond Individualized Recourse: Interpretable and Interactive Summaries of Actionable Recourses. In Adv. in Neural Information Processing Systems, NeurIPS 2502–2529.

[22] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “Why Should I Trust You?”. In Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD ’16: 1315–1324.

[23] Tim Miller. 2019. Explanation in Artificial Intelligence: Insights from the Social Sciences. Artificial Intell. 267 (6 2019), 1–38.

[24] Shubham Rathi. 2019. Generating Counterfactual and Contrastive Explanations for Tabular Data. In Proc. of The Web Conf. 2020, WWW: 3126–3132.

[25] Gregory Plumb, Denali Molitor, and Ameet S Talwalkar. 2018. Model Agnostic Supervised Local Explanations. In Proc. of the 34th AAAI Conf. on Artificial Intell., AAAI 2018: 4768–4777.

[26] Tim Morris. 2019. Explanation in Artificial Intelligence: Insights from the Social Sciences. Artificial Intell. 267 (6 2019), 1–38.

[27] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Kristijonas Čyras. 2019. Generalized Integrated Gradients: A practical method for explaining diverse machine learning models. J. of Machine Learning R. – Under Review (2019). arXiv:1909.01869

[28] Scott M. Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In Adv. in Neural Information Processing Systems, NeurIPS 4768–4777.

[29] Luke Merrick and Ankur Taly. 2020. The Explanation Game: Explaining Machine Learning Models Using Shapley Values. In Int. Cross-Domain Conf. for Machine Learning and Knowledge Extraction, CD-MAKE. 17–38.

[30] Tim Miller. 2019. Explanation in Artificial Intelligence: Insights from the Social Sciences. Artificial Intell. 267 (6 2019), 1–38.

[31] Scott M. Lundberg, Dvirat Mahajan, Chanhoo Tan, and Amit Sharma. 2021. Towards Unifying Feature Attribution and Counterfactual Explanations: Different Means to the Same End. In Proc. of the 2021 AAAI/ACM Conf. on AI, Ethics, and Society, AIES 652–663.

[32] Scott M. Lundberg, Dvirat Mahajan, Chanhoo Tan, and Amit Sharma. 2021. Towards Unifying Feature Attribution and Counterfactual Explanations: Different Means to the Same End. In Proc. of the 2021 AAAI/ACM Conf. on AI, Ethics, and Society, AIES 652–663.

[33] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. 2020. Learning Model-Agnostic Counterfactual Explanations for Tabular Data. In Proc. of The Web Conf. 2020, WWW: 3126–3132.

[34] Rafael Poyiadzi, Kaifer Sokol, Saul Santos-Rodriguez, Tijl De Bie, and Peter Flach. 2020. FACE: Feasible and Actionable Counterfactual Explanations. In Proc. of the AAAI/ACM Conf. on AI, Ethics, and Society, AIES 344–350.

[35] Rafael Poyiadzi, David Martens, Foster Provost, and Theodoros Evgeniou. 2020. A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data. SEDIC, LIME-C and SHAP-C. Adv. in Data Analysis and Classification 14 (2020), 801–819.

[36] Rafael Poyiadzi, David Martens, Foster Provost, and Theodoros Evgeniou. 2020. A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data. SEDIC, LIME-C and SHAP-C. Adv. in Data Analysis and Classification 14 (2020), 801–819.

[37] Shubham Rathi. 2019. Generating Counterfactual and Contrastive Explanations using SHAP. In 2nd Workshop on Humanizing AI (HAI) at IJCAI ’19. arXiv:1906.09293

[38] Rafael Poyiadzi, Kaifer Sokol, Saul Santos-Rodriguez, Tijl De Bie, and Peter Flach. 2020. FACE: Feasible and Actionable Counterfactual Explanations. In Proc. of the AAAI/ACM Conf. on AI, Ethics, and Society, AIES 344–350.

[39] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “Why Should I Trust You?”. In Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD ’16: 1315–1324.

[40] Chris Russell. 2019. Efficient Search for Diverse Coherent Explanations. In Proc. of the 2019 Conf. on Fairness, Accountability, and Transparency, FAccT. 20–28.

[41] Taco van Ooltrop, Ben van Dijk, and Marco Tulio Ribeiro. 2019. From local explanations to global understanding with explainable AI for trees. J. of Machine Learning R. - Under Review (2019). arXiv:1909.01869

[42] David S. Watson. 2021. Rational Shapley Values. arXiv:2011.10095

[43] Adam White and Artur d’Avila Garcez. 2019. Measurable Counterfactual Local Explanations for Any Classifier. In Proc. of the 24th European Conf. on Artificial Intell., ECAI 2529–2535.