On the Explanation of a General Decision Model

Nicolas Atienza[1,2,3] and **Christophe Labreuche**[1,2] and **Roman Bresson**[1,2]

Abstract. Feature attribution is an important class of explainable AI techniques. It only provides assessment of how much each input variables contributes to the model outcome. The Shapley value has emerged as a leading concept to ensure a fair allocation thanks to its nice property. One of its important property is efficiency. When the model is described by a directed acyclic graph, these techniques do not provide any insight on how important the intermediate nodes in the graph are. Explanation techniques based on the gradient provide influence of any node in the graph, but they do not satify any property. We introduce a game theoretical approch for explaining all nodes in a directed acyclic graph. The main idea is to see the explanation as a flow going from the input variables to the outcome through all nodes in the graph. Efficiency is generalized by a flow conservation property. We provide an instantiation of this general framework taking ideas from causality to distinguish all possible paths from a node to the top node. We compute the influence of this node in a particular path, freezing the influence of this node on the remaining paths. The Winter value, which is an extension of the Shapley value on trees, is used in this computation.

1 Introduction

Explainable AI (XAI) is an essential field of AI. It aims at providing insights to an AI expert or the end-user of what is the general behaviour of the model or why the model outputs a certain prediction for a particular instance [2]. We are interested in explaining a numerical model that is decomposed in a graph and more precisely in a Directed Acyclic Graph (DAG). Such situation is very general and encompasses Neural Networks, Decision Trees, Bayesian Networks, Multi-Criteria Decision Aiding to cite a few.

Most explanation methods ignore the graph structure and seeks to finding the features (input variables) that mostly contribute to the outcome of the model. Typical examples of this trend are LIME [19] and SHAP, which uses the Shapley value for feature attribution [23, 7, 14, 13, 15].

The interest of game-theoretical concepts such as the Shapley value is that they satisfy formal properties such as efficiency, which states that what all players earn together shall be equal to the sum of what is alloted to each player. An extension to trees has been proposed by Winter [24]. The Winter value satisfies an extended efficiency property. It says that what is alloted to a node in the tree (representing a coalition of players) shall be the sum of what is alloted to its direct predecessors in the tree. To our knowledge, there is no extension of the Shapley value on DAGs. The values existing on DAGs use the graph as a precedence among players and generate admissible coalitions accordingly.

[1] Thales Research & Technology, 91767 Palaiseau cedex, France
[2] SINCLAIR AI Lab, Palaiseau France
[3] LISN, Université Paris Saclay, France

More specific methods for Neural Network have been developped. Many methods are based on the computation of the gradient of the prediction. They can assign a relevance score to each input feature (a pixel in an image) but also to each neuron. One can mention Grad-CAM [20] and Vanilla Gradient [9], DeconvNet reverses the different steps of the neural network (filtering, pooling, activation) [25]. SmoothGrad adds noise on gradients and averages these values to smooth gradient-based explanations [22].

The drawback of gradient-based approaches is that they do not satisfy any property. We may thus obtain inconsistencies in the influences obtained with these methods among the different levels.

Our aim is to extend the Shapley value on DAGs and in particular generalize the efficiency property on DAGs. Our main idea is to see the explanation problem as a flow of the influence emerging from the input variables and going towards the prediciton node through the graph. We interpret the efficiency property as a flow conservation property saying that the sum of the (influence) incoming flow towards a node is equal to the sum of the (influence) outgoing flow leaving this node. Figure 1 illustrates what we would like to obtain. Instead of having only an influence associated to each input variable, we would like to have an influence associated to each node and edge such that flow conservation is fullfilled. For instance the influence degree $5/8$ of the green node is the sum of the influence degrees of its two outgoing edges (namely $1/4 + 3/8$). Likewise, the influence degree $1/2$ of the red node is the sum of the influence degrees of its two incoming edges (namely $3/8 + 1/8$). The general flow approach is presented in Section 3. Section 4 proposed an expression to measure the flow, based on ideas from causality and the use of the Winter value.

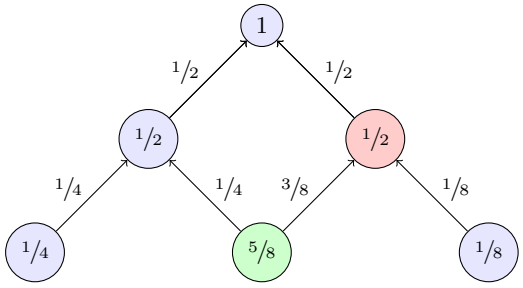

Figure 1. Example of explanation represented as a flow. A flow is associated to each node and edge such that flow conservation is satisfied.

2 Preliminaries

Consider a DAG (Directly Acyclic Graph) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is a finite set of vertices and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges. For a node

$i \in \mathcal{V}$, we denote by $\mathrm{DS}(i)$ the set of direct successors of i and by $\mathrm{DP}(i)$ the set of direct predecessors of i in this graph. The *sources* – denoted by N – are the nodes having no predecessors, and the *sinks* are the nodes having no successor. We basically wish to explain the outcome at a sink and understand how its predecessors contribute to that. W.l.o.g. we thus focus on a DAG having only one sink denoted by s.

Example 1. *Consider the DAG of Figure 2. Nodes* $1, 2, 3$ *are sources and node* 6 *is the unique sink. Moreover,* $\mathrm{DP}(4) = \{1, 2\}$ *and* $\mathrm{DS}(4) = \{6\}$. ■

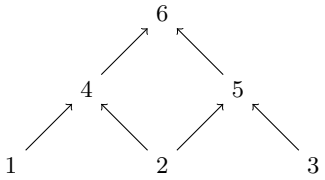

Figure 2. Example of a hierarchy of criteria.

2.1 Shapley value

The aim of Cooperative Game Theory is to determine how to share among the players the wealth obtained by all players together. The set of players is N. A *Transferable Utility (TU) game* (game in short) is a set function $v : 2^N \to \mathbb{R}$ satisfying $v(\emptyset) = 0$, where $v(S)$ for $S \subseteq N$ is the wealth produced by players S when they form a coalition.

A *value* is a function that allocates a worth to each player. It is defined as a function $\phi_i(N, v)$ representing what is given to player $i \in N$ in game v defined on the set N of players. The basic property of a value – called *efficiency* – tells that it shares the wealth $v(N)$ obtained by the grand coalition N among its members:

$$\sum_{i \in N} \phi_i(N, v) = v(N).$$

The Shapley value is a fair share of the global wealth $v(N)$ [21]:

$$\mathrm{Sh}_i(N, v) = \sum_{S \subseteq N \setminus i} \frac{(n - |S| - 1)!|S|!}{n!} \big[v(S \cup \{i\}) - v(S)\big]. \quad (1)$$

The Shapley value can also be writen as $\mathrm{Sh}_i(N, v) = \mathcal{S}_i^N(\delta_i v)$, where $\delta_i v$ is a game defined on $N \setminus \{i\}$ by $\delta_i v(S) := v(S \cup \{i\}) - v(S)$, and $\mathcal{S}_i^N$ is defined on a game v' defoned on the set $N \setminus \{i\}$ of players by $\mathcal{S}_i^N(v') = \sum_{S \subseteq N \setminus \{i\}} \mathcal{C}_a^s \, v'(S)$, with $\mathcal{C}_a^s := \frac{s!(a-s-1)!}{a!}$, and the notation that sets are denoted by capital letters (e.g. A, S) and their cardinality by the lower case letter (e.g. a, s). The Shapley value is characterized by four properties: *Additivity*, *Null player*, *Symmetry* and *Efficiency* [21].

2.2 Winter value

When the set of players is organized hierarchically with a nested coalition structure, a value shall be assigned not only to the elementary players (the leaves of the tree) but also to coalitions (the other nodes in the tree). It has been argued that the Shapley value is not consistent with the hierarchy, as the Shapley value at a coalition is not necessarily equal to the sum of the Shapley values of its members [17, 24, 12]. The Winter value has been defined to overcome this difficulty [24]. When computing the value of node $i \in N$, it is not necessary to consider the full tree and we can restrict the tree to the nodes that directly contribute to node i. More precisely, the tree is restricted to the nodes at a distance at most 1 from the path from the sink (root) of the tree to node i. The path from the root of the tree to node i is denoted by $(p_0, p_1, \ldots, p_q)$ where p_0 is the sink and $p_q = i$. For $m \in \{1, \ldots, q\}$, we set $N_m = \mathrm{DP}(p_{m-1})$ and $N'_m = N_m \setminus \{p_m\}$. Set N'_m thus contains all the siblings of node p_m.

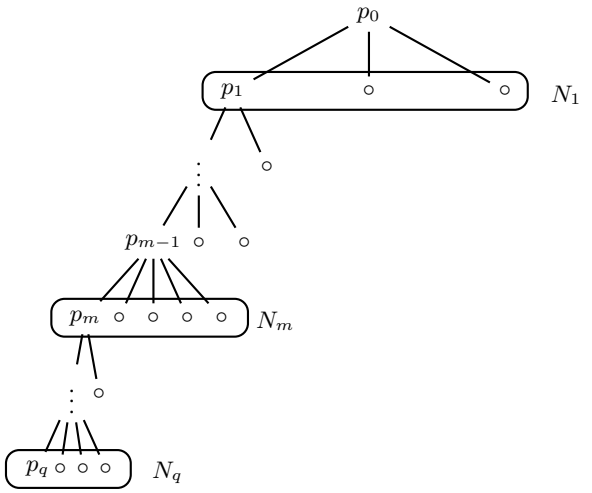

Figure 3. Restricted tree and illustration of notation p_m and N_m.

The Winter value of node i is then a recursive application of $\mathcal{S}_{p_m}^{N_m}$ at all levels $m = 1, \ldots, q$ [24, 12, 11]:

$$\mathrm{Win}_i(v) = \mathcal{S}_{p_1}^{N_1} \circ \mathcal{S}_{p_2}^{N_2} \circ \cdots \circ \mathcal{S}_{p_q}^{N_q}(\delta_i v) \quad (2)$$

$$= \sum_{S_1 \subseteq N'_1, \ldots, S_q \subseteq N'_q} \left[\prod_{m=1}^{q} \mathcal{C}_{n_m}^{s_m} \right] \times (v(S_1^q \cup \{i\}) - v(S_1^q)) \quad (3)$$

where $S_k^q := \cup_{m=k}^q S_m$.

3 Influence indices on a DAG as a flow problem

3.1 Basic Definitions

We consider the case in which each node $i \in \mathcal{V}$ is associated to a variable denoted by X_i. The domain of definition of variable X_i is denoted by $\mathcal{D}(X_i)$. A realization of variable X_i is often denoted by x_i, with $x_i \in \mathcal{D}(X_i)$. For a subset S of nodes, let $\mathcal{D}(X_S) = \times_{i \in S} \mathcal{D}(X_i)$. We assume a deterministic function providing the value of a node given the value of its direct predecessors. We assume thus that we are given a set of functions $F_i : \mathcal{D}(X_{\mathrm{DP}(i)}) \to \mathcal{D}(X_i)$ for all $i \in \mathcal{V} \setminus N$. For $x \in \mathcal{D}(X_N)$, we recursively apply the previous functions to infer the values of all variables in the DAG. More precisely, we define $x \in \mathcal{D}(X_{\mathcal{V} \setminus N})$ by $x_i = F_i(x_{\mathrm{DP}(i)})$. Finally we define the function F by $F : x_N \mapsto x_s$, which returns the value at the sink given the values at the sources.

Example 2 (Ex. 1 cont.)**.** *We have*

$$x_4 = F_4(x_1, x_2) \quad , \quad x_5 = F_5(x_2, x_3) \quad , \quad x_6 = F_6(x_4, x_5)$$

so that

$$x_6 = F(x_1, x_2, x_3) = F_6(F_4(x_1, x_2), F_5(x_2, x_3)).$$

The previous setting is very general and can be encountered in many situations. It may correspond to a Neural Network in which functions F_i are the activation functions, to a Bayesian Network in which functions F_i return the conditional probability distributions (over Boolean variables), to a hierarchical Multi-Criteria Decision Aiding model in which functions F_i are the aggregation function, to a Fuzzy Inference in which functions F_i corresponds to fuzzy and/or connectives, and so on. Lastly, in the domain of causal modeling and reasoning, the previous framework corresponds exactly to *Structural Equation Model (SEM)* [18]. In the following, we will often refer our DAG model as a SEM.

In causal reasoning, the DAG structure and the SEM allow us to make interventions on variables which are not the sources. The variables are split into two groups. The first one is the set *exogeneous* variables. Their knowledge allows us to deduce the values of all other variables. They correspond thus to the sources N. The second group is called the set of *endogeneous* variables $\mathcal{V} \setminus N$. In the framework of SEM, it is possible to reason under the *counterfactual* situation where some (endogeneous) variables in $S \subseteq \mathcal{V} \setminus N$ are (artificially) set to values y_S which are not what the SEM would give [8]. We denote by $F^{X_S \leftarrow y_S}(x)$ the value at the sink given the value x at the sources and the counterfactual values y_S. The counterfactual situation on a node overrides the evaluation of any of its predecessors.

Example 3 (Ex. 2 cont.)**.** *Assume that we wish to fix values on* $S = \{5\}$*. Then*

$$F^{X_5 \leftarrow y_5}(x_1, x_2, x_3) = F_6(F_4(x_1, x_2), y_5).$$

We note that we do not use expression $F_5(x_2, x_3)$ *to compute the value at node* 5 *as we directly have as input the value at this node.*

In order to ease the notation, we will not distinguish the initial values x at the sources N and the counterfactual values y_S. Let A be the set of nodes (endogeneous and exogeneous) on which we are given their value. These values are denoted by $x_A \in \mathcal{D}(X_A)$. We can assess the value at the sink given x_A if every path from a source to the sink insersects A. For instance, this condition is fulfilled in Ex. 1 with subsets $\{4, 5\}$ or $\{1, 2, 5\}$, but not for $\{1, 2\}$. Then we define function $F_A : \mathcal{D}(A) \to \mathcal{D}(s)$ such that $F_A(x_A)$ is the evaluation at node x_s by application of the SEM starting from the values x_A on nodes A.

Example 4 (Ex. 2 cont.)**.** *For* $A = \{1, 2, 5\}$*, we have*

$$x_6 = F_{\{1,2,5\}}(x_1, x_2, x_5) = F_6(F_4(x_1, x_2), x_5).$$

We restrict ourselves to the situation in which the sources (input variables) are Boolean – i.e. $\mathcal{D}(X_i) = \{0, 1\}$ for all $i \in N$ – but all other variables are real values $\mathcal{D}(X_i) = \mathbb{R}$ for all $i \in \mathcal{V} \setminus N$. We wish then to explain the difference of evaluation between an instance taking value 1 (say value *true*) on all sources (represented by $F(1, \ldots, 1)$ also noted $F(1_N)$) and an instance taking value 0 (say value *false*) on all sources (represented by $F(0, \ldots, 0)$ also noted $F(0_N)$).

The previous framework is quite general and does not require that the true input variables are Boolean. Indeed, feature attribution methods such as SHAP typically include a phase in which the AI model is transformed into a cooperative game, thereby requirig to binarize the variables. Let us describe shortly how this can be perfomed. Consider a function G defined on DAG $\mathcal{G}$, where the values of variables in N are real.

In local explanation, we wish to explain $G(z)$ for a particular instance $z \in \mathbb{R}^N$. A first way to binarize the set of features is to compare z to a particular reference element $r \in \mathbb{R}^N$ [15, 10]. For $x \in \mathcal{D}(X_N)$, we define the composite instance $b(z, r, x) \in \mathbb{R}^N$ by $b_i(z, r, x) = z_i$ if $x_i = 1$ and $b_i(z, r, x) = r_i$ if $x_i = 0$. Then $F(x) = G(b(z, r, x))$.

A second way is to define $F(x)$ as the conditional expected output of G when features in $S_x = \{i \in N \ : \ x_i = 1\}$ take value x [15, 10]: $F(x) = \frac{1}{K} \sum_{k=1}^{K} G(z_{S_x}, r^k_{N \setminus S_x})$, where $r^k \in \mathbb{R}^N$ (for $k = 1, \ldots, K$) are elements of a dataset, and $(z_{S_x}, r^k_{N \setminus S_x})$ is the compound option taking value of z in S_x and of r^k in $N \setminus S_x$.

3.2 Explanation problem

We wish to understand how each node in the DAG can be attributed to the difference $F(1_N) - F(0_N)$. An explanation is thus a vector assigning an influence level ϕ_i to each node i.

For a flat graph composed of one sink s, several sources and no other node, one would search for the level of contribution ϕ_i for each variable i such that the *Efficiency* property applies [21]:

$$\sum_{i \in N} \phi_i = F(1_N) - F(0_N). \tag{4}$$

If the DAG turns out to be a tree, then the efficiency is writen as a recursive property saying that the value ϕ_i alloted to a node i shall be the sum of the values of its direct predecessors [12, 24]:

$$\forall i \in \mathcal{V} \setminus N \qquad \phi_i = \sum_{j \in \mathrm{DP}(i)} \phi_j. \tag{5}$$

Note that condition (4) shall also apply.

In order to generalize efficiency on a DAG, we naturally think of a *flow diagram* in which an influence flow is assigned to each edge. The influence of edge between nodes i and j is denoted by $\varphi_{i,j}$, with $i \in \mathrm{DP}(j)$. Then the *flow conservation* property tells that the sum of all incoming flow to a node i is equal to the sum of all outgoing flow outward node i, and is also equal to the flow going through node i:

Flow Conservation (FC):

$$\forall i \in \mathcal{V} \setminus (N \cup \{s\}) \qquad \sum_{j \in \mathrm{DP}(i)} \varphi_{j,i} = \sum_{j \in \mathrm{DS}(i)} \varphi_{i,j} = \phi_i, \tag{6}$$

$$\forall i \in N \qquad \sum_{j \in \mathrm{DS}(i)} \varphi_{i,j} = \phi_i, \tag{7}$$

$$\sum_{j \in \mathrm{DP}(s)} \varphi_{j,s} = \phi_s. \tag{8}$$

FC is similar to the Kirchhoff's law in electricity. In our case, it ensure interpretability of the value assigned to the vertices and the edges.

Example 5 (Ex. 1 cont.)**.** **FC** *yields relations* $\phi_1 = \varphi_{1,4}$, $\phi_2 = \varphi_{2,4} + \varphi_{2,5}$, $\phi_3 = \varphi_{3,5}$, $\phi_4 = \varphi_{1,4} + \varphi_{2,4} = \varphi_{4,6}$, $\phi_5 = \varphi_{2,5} + \varphi_{3,5} = \varphi_{5,6}$ *and* $\phi_6 = \varphi_{4,6} + \varphi_{5,6}$. ■

General efficiency imposes that

$$\phi_s = F(1_N) - F(0_N). \tag{9}$$

4 Influence through Causal Reasoning on paths

In Ex. 1, $\varphi_{2,4}$ is the level of contribution of node 2 on sink s, through node 4. The idea would be to measure the influence on node 6 when going from 0 to 1 at node 2, for all possible values on the other nodes 1 and 3. The issue is that for $\varphi_{2,4}$, changing x_2 from 0 to 1 shall not modify the value at node 5. However, the basic way is to compute the weighted average of differences

$$F(x_1, 1_2, x_3) - F(x_1, 0_2, x_3)$$
$$= F_6(F_4(x_1, 1_2), F_5(1_2, x_3)) - F_6(F_4(x_1, 0_2), F_5(0_2, x_3))$$

over all possible values of x_1, x_3. We see that the values $F_5(1_2, x_3)$ and $F_5(0_2, x_3)$ on node 5 is not kept identical when the value at node 2 is changed. Hence the previous difference does not compute the influence of node 2 through node 4.

4.1 Decomposition on the paths through the sink

To compute the influence $\varphi_{i,j}$, the idea is to decompose the influence over the paths in DAG $\mathcal{G}$. We denote by $\mathcal{P}_i$ the set of paths going from i to the sink s, and by $\mathcal{P}_{i,j}$ the elements of $\mathcal{P}_i$ going through j, where $j \in \mathrm{DS}(i)$.

We introduce $\phi_i^p(\mathcal{G}, F)$ as the level of contribution of variable i on sink s following path $p \in \mathcal{P}_i$. Then

$$\phi_i = \sum_{p \in \mathcal{P}_i} \psi_i^p(\mathcal{G}, F) \tag{10}$$

$$\varphi_{i,j} = \sum_{p \in \mathcal{P}_{i,j}} \psi_i^p(\mathcal{G}, F) \tag{11}$$

Lemma 1. *For every node i, we have*

$$\sum_{j \in \mathrm{DS}(i)} \varphi_{i,j} = \phi_i.$$

Proof. We have

$$\sum_{j \in \mathrm{DS}(i)} \varphi_{i,j} = \sum_{j \in \mathrm{DS}(i)} \sum_{p \in \mathcal{P}_{i,j}} \psi_i^p(\mathcal{G}, F) = \sum_{p \in \mathcal{P}_i} \psi_i^p(\mathcal{G}, F) = \phi_i.$$

■

4.2 Use of the Winter value

Let $p \in \mathcal{P}_i$. We write $p = (p_1, p_2, \ldots, p_m)$ with $i = p_1$ and $s = p_m$. We define a retricted rooted tree $\mathcal{T}_{i,p} = (\mathcal{V}', \mathcal{E}')$ with root s, where

- $\mathcal{V}'$ is composed of the elements $\{p_2, \ldots, p_m\}$ and all direct predecessors of these vertices in $\mathcal{G}$;
- $\mathcal{E}' = \{(i_1, i_2) \in \mathcal{E} : i_1, i_2 \in \mathcal{V}', i_2 \in p \text{ and } [i_1 \notin p \text{ or } i_1 \text{ is direct predecessor of } i_2 \text{ in } p]\}$.

We remove from $\mathcal{G}$ the edges which would make $\mathcal{T}_{i,p}$ a DAG and not a tree. This allows us to make safe interventions.

Example 6 (Ex. 5 cont.)**.** *We have the following paths:*

$\mathcal{P}_1 = \{p_1^1\}$ *with* $p_1^1 = (1, 4, 6)$

$\mathcal{P}_2 = \{p_2^1, p_2^2\}$ *with* $p_2^1 = (2, 4, 6)$ *and* $p_2^2 = (2, 5, 6)$

$\mathcal{P}_3 = \{p_3^1\}$ *with* $p_3^1 = (3, 5, 6)$

$\mathcal{P}_4 = \{p_4^1\}$ *with* $p_4^1 = (4, 6)$

$\mathcal{P}_5 = \{p_5^1\}$ *with* $p_5^1 = (5, 6)$

Fig. 4-Left is the restricted tree corresponding to p_1^1 and p_2^1. Fig. 4-Right is the restricted tree corresponding to p_2^2 and p_3^1. Fig. 5 is the restricted tree corresponding to p_4^1 and p_5^1. ■

Lemma 2. *$\mathcal{T}_{i,p}$ is a rooted tree*

Proof. *Consider $i_1, i_2 \in \mathcal{V}'$.*

We remove from $\mathcal{G}$ the edges from two elements of p which are not direct successor in p. This avoids having multiple paths between two nodes in $\mathcal{T}_{i,p}$. Hence $\mathcal{T}_{i,p}$ is a rooted tree. ■

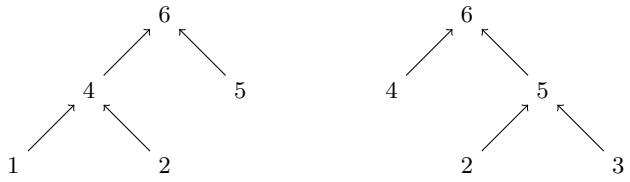

Figure 4. The left figure is the restricted tree corresponding to p_1^1 and p_2^1; The right figure is the restricted tree corresponding p_2^2 and p_3^1.

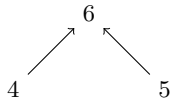

Figure 5. Rooted trees for paths p_4^1 and p_5^1.

We wish to define the influence of node i in path p – denoted by $\psi_i^p(\mathcal{G}, F)$.

We set $N_k' = \mathrm{DP}(p_{k+1}) \setminus \{p_k\}$ for any $k \in \{1, \ldots, m-1\}$. The Winter value of node i is defined by [24, 12, 11]:

$$\mathrm{Win}_i(\mathcal{T}_{i,p}, v) = \sum_{S_{m-1} \subseteq N_{m-1}', \ldots, S_1 \subseteq N_1'} \left[\prod_{k=1}^{m-1} \mathcal{C}_{n_k'+1}^{s_k} \right] \times (v(S_1^q \cup \{i\}) - v(S_1^q)) \tag{12}$$

where $S_k^q := \cup_{l=k}^q S_l$ and $\mathcal{C}_a^s := \frac{s!(a-s-1)!}{a!}$. Let $A_i := {N'}_1^{m-1} \cup \{i\}$ be the leaves of rooted tree $\mathcal{T}_{i,p}$. Game v is defined by $v(S) = F_{A_i}(1_S, 0_{A_i \setminus S})$.

Then

$$\psi_i^p(\mathcal{G}, F) = \mathrm{Win}_i(\mathcal{T}_{i,p}, v), \tag{13}$$

where Win denoted the Winter value.

Example 7 (Ex. 6 cont.)**.** *We obtain*

$$\begin{aligned}\psi_1^{p_1^1}(\mathcal{G},F) &= \mathrm{Win}_1(\mathcal{T}_{1,p_1^1},v) = \frac{v(1)}{4} + \frac{v(12)-v(2)}{4} \\ &\quad + \frac{v(15)-v(5)}{4} + \frac{v(125)-v(25)}{4} \\ &= \frac{F_{\{1,2,5\}}(1_1,0_2,0_5) - F_{\{1,2,5\}}(0_1,0_2,0_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,1_2,0_5) - F_{\{1,2,5\}}(0_1,1_2,0_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,0_2,1_5) - F_{\{1,2,5\}}(0_1,0_2,1_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,1_2,1_5) - F_{\{1,2,5\}}(0_1,1_2,1_5)}{4}\end{aligned}$$

$$\begin{aligned}\psi_2^{p_2^1}(\mathcal{G},F) &= \mathrm{Win}_2(\mathcal{T}_{2,p_2^1},v) = \frac{v(2)}{4} + \frac{v(12)-v(1)}{4} \\ &\quad + \frac{v(25)-v(5)}{4} + \frac{v(125)-v(15)}{4} \\ &= \frac{F_{\{1,2,5\}}(0_1,1_2,0_5) - F_{\{1,2,5\}}(0_1,0_2,0_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,1_2,0_5) - F_{\{1,2,5\}}(1_1,0_2,0_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(0_1,1_2,1_5) - F_{\{1,2,5\}}(0_1,0_2,1_5)}{4} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,1_2,1_5) - F_{\{1,2,5\}}(1_1,0_2,1_5)}{4}\end{aligned}$$

$$\begin{aligned}\psi_2^{p_2^2}(\mathcal{G},F) &= \mathrm{Win}_2(\mathcal{T}_{2,p_2^2},v) = \frac{v(2)}{4} + \frac{v(23)-v(3)}{4} \\ &\quad + \frac{v(24)-v(4)}{4} + \frac{v(234)-v(34)}{4} \\ &= \frac{F_{\{2,3,4\}}(1_2,0_3,0_4) - F_{\{2,3,4\}}(0_2,0_3,0_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,1_3,0_4) - F_{\{2,3,4\}}(0_2,1_3,0_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,0_3,1_4) - F_{\{2,3,4\}}(0_2,0_3,1_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,1_3,1_4) - F_{\{2,3,4\}}(0_2,1_3,1_4)}{4}\end{aligned}$$

$$\begin{aligned}\psi_3^{p_3^1}(\mathcal{G},F) &= \mathrm{Win}_3(\mathcal{T}_{3,p_3^1},v) = \frac{v(3)}{4} + \frac{v(23)-v(2)}{4} \\ &\quad + \frac{v(34)-v(4)}{4} + \frac{v(234)-v(24)}{4} \\ &= \frac{F_{\{2,3,4\}}(0_2,1_3,0_4) - F_{\{2,3,4\}}(0_2,0_3,0_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,1_3,0_4) - F_{\{2,3,4\}}(1_2,0_3,0_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(0_2,1_3,1_4) - F_{\{2,3,4\}}(0_2,0_3,1_4)}{4} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,1_3,1_4) - F_{\{2,3,4\}}(1_2,0_3,1_4)}{4}\end{aligned}$$

$$\begin{aligned}\psi_4^{p_4^1}(\mathcal{G},F) &= \mathrm{Win}_4(\mathcal{T}_{4,p_4^1},v) = \frac{v(4)}{2} + \frac{v(45)-v(5)}{2} \\ &= \frac{F_{\{4,5\}}(1_4,0_5) - F_{\{4,5\}}(0_4,0_5)}{2} \\ &\quad + \frac{F_{\{4,5\}}(1_4,1_5) - F_{\{4,5\}}(0_4,1_5)}{2}\end{aligned}$$

$$\begin{aligned}\psi_5^{p_5^1}(\mathcal{G},F) &= \mathrm{Win}_5(\mathcal{T}_{5,p_5^1},v) = \frac{v(5)}{2} + \frac{v(45)-v(4)}{2} \\ &= \frac{F_{\{4,5\}}(0_4,1_5) - F_{\{4,5\}}(0_4,0_5)}{2} \\ &\quad + \frac{F_{\{4,5\}}(1_4,1_5) - F_{\{4,5\}}(1_4,0_5)}{2}\end{aligned}$$

One can easily check that

$$\begin{aligned}&\psi_1^{p_1^1}(\mathcal{G},F) + \psi_2^{p_2^1}(\mathcal{G},F) \\ &= \frac{F_{\{1,2,5\}}(1_1,1_2,0_5) - F_{\{1,2,5\}}(0_1,0_2,0_5)}{2} \\ &\quad + \frac{F_{\{1,2,5\}}(1_1,1_2,1_5) - F_{\{1,2,5\}}(0_1,0_2,1_5)}{2} \\ &= \psi_4^{p_4^1}(\mathcal{G},F)\end{aligned}$$

and

$$\begin{aligned}&\psi_2^{p_2^2}(\mathcal{G},F) + \psi_3^{p_3^1}(\mathcal{G},F) \\ &= \frac{F_{\{2,3,4\}}(1_2,1_3,0_4) - F_{\{2,3,4\}}(0_2,0_3,0_4)}{2} \\ &\quad + \frac{F_{\{2,3,4\}}(1_2,1_3,1_4) - F_{\{2,3,4\}}(0_2,0_3,1_4)}{2} \\ &= \psi_5^{p_5^1}(\mathcal{G},F)\end{aligned}$$

■

The following example illustrates the previous formulas.

Example 8 (Ex. 9 cont.)**.** *Consider functions*

$$\begin{aligned}F_4(x_1,x_2) &= \min(x_1,x_2) \\ F_5(x_2,x_3) &= \frac{3\,x_2 + x_3}{4} \\ F_6(x_4,x_5) &= x_4 \times x_5\end{aligned}$$

We obtain

$$\begin{aligned}\psi_1^{p_1^1}(\mathcal{G},F) &= \frac{1}{4} \\ \psi_2^{p_2^1}(\mathcal{G},F) &= \frac{1}{4} \\ \psi_2^{p_2^2}(\mathcal{G},F) &= \frac{3}{8} \\ \psi_3^{p_3^1}(\mathcal{G},F) &= \frac{1}{8} \\ \psi_4^{p_4^1}(\mathcal{G},F) &= \frac{1}{2} \\ \psi_5^{p_5^1}(\mathcal{G},F) &= \frac{1}{2}\end{aligned}$$

The influence degrees at the vertices and edges are directly derived and are displayed in Figure 1.

We note some close relationship between ϕ_i and ATE (average treatment effect):

$$\begin{aligned}\mathrm{ATE} &= \mathbb{E}_X[\mathrm{CATE}(X)] \\ &= \mathbb{E}_X\left[\mathbb{E}_{Y|X}(Y|\mathrm{do}(T=1),X) - \mathbb{E}_{Y|X}(Y|\mathrm{do}(T=0),X)\right] \\ &= \mathbb{E}_X\left[Y_X(\mathrm{do}(T=1)) - Y_X(\mathrm{do}(T=0))\right]\end{aligned}$$

The last inequality holds if we can directly perform interventions on the model, which is the case when we have a SEM. ϕ_i is the counterpart of ATE; setting variable i to 1 or 0 is the counterpart of intervention $\mathrm{do}(T=1)$ or $\mathrm{do}(T=0)$; the average in the Winter value (which is a simple average over the permutations compatible with $\mathcal{T}_{i,p}$) and over the paths in $\mathcal{P}_i$ is the counterpart of $\mathbb{E}_X$.

Lemma 3. *Proposal defined by (10) and (11) satisfies to (6), (7), (8) and (9).*

Proof. By Lemma 1, we have $\sum_{j\in\mathrm{DS}(i)} \varphi_{i,j} = \phi_i$.

Moreover

$$\sum_{j\in\mathrm{DP}(i)} \varphi_{j,i} = \sum_{j\in\mathrm{DP}(i)} \sum_{p\in\mathcal{P}_{j,i}} \psi_j^p(\mathcal{G},F).$$

One can readily see that $\mathcal{P}_{j,i} = \{(j,p) \,:\, p\in\mathcal{P}_i\}$, where (j,p) is the path starting from j and then following p. Hence

$$\sum_{j\in\mathrm{DP}(i)} \varphi_{j,i} = \sum_{p\in\mathcal{P}_i} \sum_{j\in\mathrm{DP}(i)} \psi_j^{(j,p)}(\mathcal{G},F).$$

From the efficiency property of the Winter value, $\sum_{j\in\mathrm{DP}(i)} \psi_j^{(j,p)}(\mathcal{G},F) = \psi_i^p(\mathcal{G},F)$. Hence

$$\sum_{j\in\mathrm{DP}(i)} \varphi_{j,i} = \sum_{p\in\mathcal{P}_i} \psi_i^p(\mathcal{G},F) = \phi_i.$$

Hence (6) is proved.

We obtain (7) and (8) in a similar way.

Relation (9) follows from the general efficiency of the Winter value. ■

Example 9. *Consider the graph of Figure 6. For node* 4*, we have*

$$\begin{aligned}\phi_4 &= \varphi_{1,4} + \varphi_{2,4} \\ \varphi_{1,4} &= \psi_1^{(1,4,7,9,11)} + \psi_1^{(1,4,7,10,11)} \\ \varphi_{2,4} &= \psi_2^{(2,4,7,9,11)} + \psi_2^{(2,4,7,10,11)}\end{aligned}$$

For node 7*, we have*

$$\begin{aligned}\phi_7 &= \varphi_{4,7} + \varphi_{5,7} \\ \varphi_{4,7} &= \psi_4^{(4,7,9,11)} + \psi_4^{(4,7,10,11)} \\ \varphi_{5,7} &= \psi_5^{(5,7,9,11)} + \psi_5^{(5,7,10,11)}\end{aligned}$$

By Efficiency of the Winter value,

$$\begin{aligned}\phi_1^{(1,4,7,9,11)} + \phi_2^{(2,4,7,9,11)} &= \phi_4^{(4,7,9,11)} \\ \phi_1^{(1,4,7,10,11)} + \phi_2^{(2,4,7,10,11)} &= \phi_4^{(4,7,10,11)}\end{aligned}$$

■

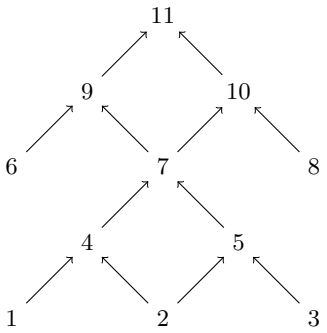

Figure 6. Example of a hierarchy of criteria.

5 Related Work

The Shapley values are widely used (especially in Machine Learning) to locally explain a prediction model [14]. The potential hierarchical of DAG structure is ignored. These values have been generalized to hierarchical models [12], and can serve as both local (explanation for a particular instance) and global (general interpretation of the model) explanation. The extensions of the Shapley value in Cooperative Game Theory on graph structures. The Shapley value is approximated by restricting coalitions to the neighbor features in the graph of conditional dependencies among features [6]. In another approach, the structure of conditional dependencies among the features is represented in a graph, and the Shapley value can then be approximated by restricting coalitions to the neighbor features in the graph [6]. The Shapley value can then be approximated by restricting coalitions to the neighbor features in the graph structure, which yields a generalization of the Myerson value [16].

In other domains, some indices have been defined on DAGs. One can mention centrality indices in social network [4, 5], and the evaluation of the strength of arguments in argumentation graphs [1]. PageRank defines a score assigned to all web sites measuring their popularity [3]. For a node i, it is defined by

$$\mathrm{PR}(i) = (1-d) + d\sum_{j\in\mathrm{Ch}(i)} \frac{\mathrm{PR}(j)}{|\mathrm{Pa}(j)|},$$

where p (damping factor, say 0.85) is the likelihood of choosing a random link from the page that is currently visited. We sum over all websites $\mathrm{DP}(i)$ referencing i, and those websites are discounted by the number of webpages $|\mathrm{DP}(j)|$ they reference. However, these works do not seek to share a global contribution and are not applicable to the computaiton of the importance of a variable.

6 Conclusion

We have proposed a novel approach for explaining a model computed on a DAG, extending the game theoretical approaches using the Shapley value. The efficiency property of the Shapley value is interpreted as a flow conservation property where the explanation takes the form of a flow assigned on each arc and node. An instanciation is proposed, using the Winter value.

REFERENCES

[1] L. Amgoud, J. Ben-Naim, and S. Vesic, 'Measuring the intensity of attacks in argumentation graphs with Shapley value', in *Proceedings of*

the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17), pp. 63–69, Melbourne, Australia, (August 2017).
[2] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, 'Explainable artificial intelligence (XAI) : Concepts, taxonomies, opportunities and challenges toward responsible AI', *Information Fusion*, **58**, 82–115, (2020).
[3] S. Bin and L. Page, 'The anatomy of a large-scale hypertextual web search engine', Technical report, Stanford University, (1998).
[4] P. Bonacich, 'Power and centrality: A family of measures', *American Journal of Sociology*, **92**(5), 1170–1182, (1987).
[5] S. Borgatti, 'Centrality and network flow', *Social Networks*, **27**, 55–71, (2005).
[6] J. Chen, L. Song, M. Wainwright, and M. Jordan, 'L-Shapley and C-Shapley: Efficient Model Interpretation for Structured Data', in *arXiv preprint arXiv:1808.02610*, (2018).
[7] A. Datta, S. Sen, and Y. Zick, 'Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems', in *IEEE Symposium on Security and Privacy*, San Jose, CA, USA, (May 2016).
[8] Joseph Y Halpern and Judea Pearl, 'Causes and Explanations: A Structural-Model Approach - Part I: Causes', in *Proceedings of the Seventeenth Conference on Uncertainy in Artificial Intelligence (UAI)*, pp. 194–202, San Francisco, CA, (June 2001).
[9] S. Karen, A. Vedaldi, and A. Zisserman, 'Deep inside convolutional networks: Visualising image classification models and saliency maps', in *arXiv:1312.6034*, (2013).
[10] I. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler, 'Problems with Shapley-value-based explanations as feature importance measures', in *37th International Conference on Machine Learning (ICML'2020)*, 5491–5500, (2020).
[11] C. Labreuche, 'Explaining hierarchical multi-linear models', in *Proceedings of the 13th international conference on Scalable Uncertainty Management (SUM 2019)*, Compiègne, France, (December 2019).
[12] C. Labreuche and S. Fossier, 'Explaining multi-criteria decision aiding models with an extended shapley value', in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI 2018)*, pp. 331–339, Stockholm, Sweden, (July 2018).
[13] S. Lundberg, G. Enrion, and S.I. Lee, 'Consistent individualized feature attribution for tree ensembles', in *arXiv preprint arXiv:1802.03888*, (2018).
[14] S. Lundberg and S.I. Lee, 'A Unified Approach to Interpreting Model Predictions', in *31st Conference on Neural Information Processing Systems (NIPS 2017)*, eds., I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 4768–4777, Long Beach, CA, USA, (2017).
[15] L. Merrick and A. Taly, 'The Explanation Game: Explaining Machine Learning Models with Cooperative Game Theory', in *arXiv preprint arXiv:1909.08128*, (2018).
[16] R. Myerson, 'Graphs and cooperation in games', *Math. of Operation Research*, **2**, 225–229, (1977).
[17] G. Owen, 'Values of games with a priori unions', in *Essays in Mathematical Economics and Game Theory*, ed., O. Moeschlin R. Hein, 76–88, Springer Verlag, (1977).
[18] J. Pearl, *Causality: Models, Reasoning and Inference*, Cambridge University Press, 2000.
[19] M.T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?": Explaining the Predictions of Any Classifier', in *KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, San Francisco, California, USA, (2016).
[20] R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, 'Grad-cam: Visual explanations from deep networks via gradient-based localization', in *arXiv:1610.02391*, (2016).
[21] L. S. Shapley, 'A value for n-person games', in *Contributions to the Theory of Games, Vol. II*, eds., H. W. Kuhn and A. W. Tucker, number 28 in Annals of Mathematics Studies, 307–317, Princeton University Press, (1953).
[22] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, 'Smoothgrad: removing noise by adding noise', in *arXiv:1706.03825*, (2017).
[23] E. Štrumbelj and I. Kononenko, 'An efficient explanation of individual classifications using game theory', *Journal of Machine Learning Research*, **11**, 1–18, (2010).
[24] E. Winter, 'A Value for Cooperative Games with Levels Structure of Cooperation', *Int. J. of Game Theory*, **18**, 227–240, (1989).
[25] M. Zeiler and F. Rob, 'Visualizing and understanding convolutional networks', in *European conference on computer vision (ECCV)*, pp. 818–833, Zurich, Switzerland, (September 2014).