Explanation of Machine Learning Outcomes through Argumentation

Stefano Bistarelli, Alessio Mancinelli, Francesco Santini, Carlo Taticchi
Università degli Studi di Perugia
[name.surname]@unipg.it

Abstract

The requirement of explainability is gaining more and more importance in Artificial Intelligence applications based on Machine Learning techniques, especially in those contexts where critical decisions are entrusted to software systems (think, for example, of financial and medical consultancy). Making the results of such applications understandable by human users not only increases confidence in Artificial Intelligence in general but also allows one to obtain insights into the functioning of the model used and to design debugging operations. In this paper, we propose an argumentation-based methodology for explaining the results predicted by Machine Learning models. Argumentation provides frameworks that can be used to represent and analyse logical relations between pieces of information, serving as a basis for constructing human tailored rational explanations to a given problem.

1 Introduction

The term Explainable Artificial Intelligence (XAI) refers to the principle by which the operating procedures and the results offered by intelligent computer systems are made understandable to human users. This concept has been studied for decades and have seen unprecedented growth in recent years [Arrieta *et al.*, 2020]. The black box model used in Machine Learning (ML) is considered one of the major problems in the application of Artificial Intelligence (AI) techniques [von Eschenbach, 2021]: it makes machine decisions non-transparent and often incomprehensible even to experts or developers themselves, which reduces trust in ML and AI in general. The need for explainability is exacerbated in critical context where the decisions made have a direct impact on people's lives (e.g., financial plans or disease treatments). Understanding the choices made by AI algorithms is therefore of fundamental importance, not only to increase trust in AI, but also to provide insights into the model itself and to carry out debugging operations [Kulesza *et al.*, 2015]. Another reason for the strong interest in understanding the processes behind ML algorithms is the increase in public sensitivity towards privacy. The introduction of various international legislation, such as the GPDR[1], imposes high levels of privacy protection and, at the same time, transparency in the processing of information, which AI algorithms currently struggle to achieve.

Among the various approaches to explanation, argumentative models play a fundamental role both in the literature relating to AI and in the social sciences, given their dialectical nature which allows linking applications to the human beings who develop and use them [Cyras *et al.*, 2021]. The use of the argumentation in XAI is supported by the solid foundation and flexibility provided by the wide variety of frameworks offered in the literature. Abstract Argumentation Frameworks (AFs) [Dung, 1995] allow for specifying arguments and dialectical relations between them. Such frameworks are also endowed with a set of semantics for evaluating the acceptability of arguments themselves.

In this paper, we provide an argumentative interpretation of both the training process and the results predicted by Machine Learning models. We take in input a dataset characterised by a certain number of features, one of which represents the class of the record, and we build an AF whose arguments consist of a subset of selected features related to each other in accordance with their correlation value. We then use argumentation semantics to evaluate the acceptability of the arguments in the BAF: starting from justified arguments, we can build an explanation tree that shows motivations behind the attribution of a certain class to a given record.

2 Preliminaries

In his seminal paper [Dung, 1995], Dung defines the building blocks of abstract argumentation: an Abstract Argumentation Framework is a pair $\langle Arg, R\rangle$ where Arg is a set of arguments and R is a binary attack relation on Arg. For two arguments $a, b \in Arg$, $(a, b) \in R$ represents an attack directed from a against b. Moreover, we say that an argument b is *defended* by a set $B \subseteq Arg$ if and only if, for every argument $a \in Arg$, if $R(a, b)$ then there is some $c \in B$ such that $R(c, a)$. A generalisation of AFs is provided by Bipolar Argumentation Frameworks [Cayrol e Lagasquie-Schiex, 2005], which admit two different types of relations between arguments: an attack relation and a support one. Formally, a Bipolar Argumentation Framework is a tuple $\langle Arg, R_{att}, R_{supp}\rangle$ where

[1] General Data Protection Regulation: https://gdpr-info.eu/.

Arg is again the set of arguments, while R_{att} and R_{supp} are two binary relation on Arg representing attacks and supports, respectively.

The goal is to establish which are the acceptable arguments according to a certain semantics, namely a selection criterion. Non-accepted arguments are rejected. Different kinds of extension-based semantics (e.g., admissible, complete, stable, semi-stable, preferred, and grounded) have been introduced [Dung, 1995; Baroni *et al.*, 2011] that reflect qualities which are likely to be desirable for "good" subsets of arguments. In particular, the semi-stable semantics is particularly suitable for constructing explanations: it provides a solid justification for accepted arguments [Baroni *et al.*, 2011], and always admit a non-empty set of extensions.

3 Explanation through Argumentation

The proposed approach consists of the following steps, which, staring from an input dataset composed of n records, allows to build a BAF showing the dialectical reasoning behind the assignment of a certain class to a given record.

1. **Dataset Clustering.** Starting from the input dataset, we create a new clustered dataset in which numerical features are split into categories that group ranges of values. This allows us to obtain a more appropriate and concise explanation.
2. **BAF Generation.** We build a BAF based on the correlation matrix computed among the features.
3. **Breaking BAF's Complete Symmetry.** Given the correlation matrix, we apply a procedure that breaks the complete symmetry so as to extract causality.
4. **Computing Extensions and Explanation Trees.** We compute the semi-stable extensions of the previously obtained framework and we build the explanation tree for the selected class.

A detailed description of each step follows.

3.1 Dataset Clustering

We begin by selecting a dataset to analyse, together with the problem's class and a list of categorical and numerical features. For each possible value of categorical features a new column is generated in the clustered dataset. For instance, if the feature *A* can take three values 0, 1 and 2, we add three new columns, respectively for *A=0*, *A=1* and *A=2*. If in a certain record of the original dataset the feature *A* takes value 0, then the corresponding record *A=0* in the clustered dataset is set to 1, while *A=1* and *A=2* are both set to 0.

Generating a new column for every possible value is not feasible, instead, for numerical features. In this case, we find the best *split* based on entropy [Fayyad e Irani, 1993], also taking into account the class specified by the user. The generation of new records with values of either 1 or 0, occurs in the same way as described for categorical features.

3.2 BAF Generation

Following the features *splitting* phase, a BAF is generated starting from a subset of arguments, chosen from the list of features, that will be used for the explanation. The correlation matrix can be computed by using rank coefficients as the Kendall [Kendall, 1974] one.

We start building the BAF by adding an attack with weight equal to -1 between all the categorical and numerical features generated from the splitting of a same feature. For example, if the features B, C and D are created by splitting the feature A, then we add symmetrical attacks with weight -1 between arguments B, C and D. In this way, we prevent arguments coming from the same feature to be in the same extension. Then, to determine what kind of relation (between support and attack) exists between two arguments A and B, we look at their correlation value. If such a value is negative, we add an attack between A and B; if it is positive, we add a support relation. In both cases, the weight of the relation is equal to the correlation value between A and B. At this stage, all the relations in such an assembled BAF, regardless of their type, are symmetrical.

3.3 Breaking BAF's Complete Symmetry

Breaking the symmetry of the obtained framework is crucial to know which of two related features implies the other. Such a causal relation cannot be studied only relying on the correlation matrix (which is symmetrical by construction), hence we consider the conditional probability [Borovcnik, 2012] between the features. Given two features A and B, we compute the conditional probability of A given B (written $P(A|B)$) and the conditional probability of B given A ($P(B|A)$). If $P(A|B) > P(B|A)$, then the relation between B and A is removed[2], since A is more probable to happen. In the opposite case, we would have removed the relation from A to B. Note that we only remove relations between arguments that do not come from the same feature. Indeed, we do not want to remove the symmetrical attack with weight -1 we added in the previous step between arguments obtained through a *split*.

3.4 Computing Extensions and Explanation Trees

We are now able to compute the set of acceptable arguments. In our BAF, each relation between two features A and B is endowed with a probability corresponding to the value in the correlation matrix between A and B. Such a probability represents uncertainty over the topology of the graph. We use the constellation approach [Li *et al.*, 2011] to compute the probability of a set of arguments of being a semi-stable extension[3]: this gives an idea of the plausibility of each possible explanation.

Finally, we build the explanation tree. We show an example in Figure 1 in which the root node A represents the assignment of the class we want to explain. We can add to the tree nodes that attack/support the root: the explanation can be produced by showing features that either support the class assignment or are against it or both. In Figure 1, the root

[2]We also consider a threshold not to remove features with conditional probabilities that are too similar to each other.

[3]Due to computational issues, we use a workaround to decrease the complexity of the operation (see Section 4 for implementation details.)

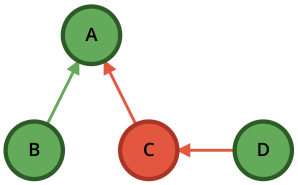

Figura 1: Example of explanation tree.

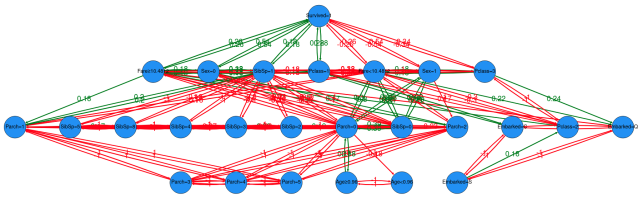

Figura 2: BAF generated for the class *Survived=1* of the Titanic dataset.

has a supporting feature B, which must be an accepted argument belonging to an extension found in the previous step. Argument C, instead, is attacking A and must be defeated by another argument supporting the root (D in our case).

4 Experiments

We now present an example of explanation obtained through the procedure described in Section 3 and using the "Titanic" dataset[4], which contains records relating to people involved in the Titanic disaster (see Table 1 for the list of features). The class to predict is *Survived*, which determines whether a person survived the disaster (value 1) or not (value 0).

Feature	Possible Values	Type
Pclass	$1, 2, 3$	categorical
sex	$0, 1$	categorical
SibSp	$0, 1, 2, 3, 4, 5, 6, 7, 8$	categorical
Parch	$0, 1, 2, 3, 4, 5, 6$	categorical
Embarked:	C, Q, S	categorical
Age	$0.17 - 76$	numerical
Fare	$0 - 512$	numerical

Tabella 1: Titanic dataset features.

First, we select the problem class (for which we want to build an explanation), the categorical features and the numerical ones. We want to find an explanation for the class *Survived=1* using all the dataset features except *Survived=0*. Then, we compute the correlation matrix for the selected features and obtain a BAF with only symmetrical relations. The symmetry is broken through the conditional probability computed for arguments which attack/support each other. When the difference in conditional probability is minimal, however, we want to to maintain both relations, because there is not enough confidence in determining which feature implies the other. Hence we specify a correlation threshold which must be reached in order to remove one of the two symmetrical relations. If not, both relations remain in the BAF. In this example, we use a correlation threshold of 0.17. A percentage threshold is also used to manipulate the number of attacks and supports to remove. Let m and n be the number of records that have $A = 1$ and $B = 1$, respectively. With a threshold of $x\%$, the relation between B and A is removed only if the condition $\frac{m \cdot x}{100} > n$ is satisfied. We choose the minimum values possible that keep the graph connected, which is, in this case, a removal percentage of 30%. As we can see in Figure 2, we obtain a BAF with 26 arguments and 179 relations (131 attacks and 48 supports) within a single connected component. The "main" argument visualised at the top of the BAF is that representing the assignment of the class *Survived=1*.

To identify the set of arguments which are more likely to be accepted, we need to compute the semi-stable extensions and find their probability of being admissible. For obtaining the list of semi-stable extensions, we first need to translate the considered BAF into a classical AF where only attack relations are allowed [Boella *et al.*, 2010]. The translation phase begins with the **support relations closure**: given three arguments A, B and C, if $supp(A, B) = x$ and $supp(B, C) = y$, we add a support relation between A and C such that $supp(A, C) = x * y$. The next step is the **attack relations closure** and comprises two distinct phases. First, we look for triples of arguments A, B and C with $supp(A, B) = x$ and $att(B, C) = y$, and we add an attack relation between A and C such that $att(A, C) = x * y$. Then, for all arguments A, B and C with $att(A, B) = x$ and $supp(C, B) = y$, and we add an attack $att(C, A) = -y$. At this point, we delete all the support relations from the modified BAF, thus obtaining a classical AF [Boella *et al.*, 2010]. After the transitive closure and the removal of supports, we obtain an AF with 265 attack relations for which we compute the set of semi-stable extensions. Then, we use the tool described in [Bistarelli *et al.*, 2018] to compute the probability of each semi-stable extension found to be also admissible.

For instance, Extension (1) is a semi-stable extension of the generated AF, which is also an admissible extension with probability 1 (the highest possible) and contains the argument *Survived=1*.

$$\begin{array}{c} \textit{Fare}{\geq}\textit{10.4812}, \textit{Age}{<}\textit{0.96}, \textbf{\textit{Survived=1}}, \\ \textit{Embarked=C}, \textit{Sex=0}, \textit{Parch=1}, \textit{SibSp=1}, \textit{Pclass=1} \end{array} \quad (1)$$

Extension (1) represent a good explanations of why the individual survives, since, being semi-stable, it provides the maximal number of arguments justifying the class *Survived=1*. Finally, starting from arguments of the selected extension, we produce the explanation tree of Figure 3, where accepted arguments are labelled in and highlighted in green, while rejected ones are labelled out and highlighted in red. Possible $undec$ arguments (not present in our example) would have been removed as they are not helpful in the explanation.

Looking at the obtained explanation we can conclude, for instance, that the person in question survived because "she is a woman (*Sex=0*), with a paid ticket (*Fare*$\geq$*10.4812*) and travelling first class (*Pclass=1*)". Indeed, arguments representing those features in Figure 3 attack other arguments that

[4]The dataset is taken from https://www.kaggle.com/.

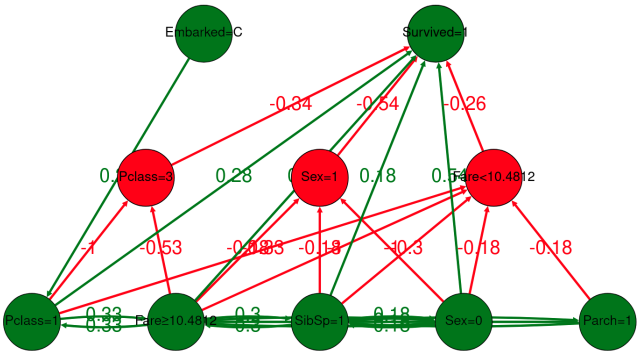

Figura 3: An explanation tree for the class *Survived=1* of the Titanic dataset.

are against the assignment of the class *Survived=1*, standing in turn for being male (*Sex=1*) and having a third-class ticket (*Pclass=3*) with a low fare (*Fare<10.4812*). From Figure 3 we could also assume that the passengers embarked at Cherbourg mainly had first class tickets, indeed argument *Embarked=C* supports *Pclass=1*.

5 Conclusion and Future Work

With this work, we provide an argumentative interpretation of the answers provided by Machine Learning models, proposing AFs to obtain a dialectical explanation. To this end, we devised a procedure which allows the construction of a BAF and an explanation tree for each computed semi-stable extension. Such a tree represents the *dialectical reasoning* among the features of the analysed problem, and together with the produced BAF, explains why a certain class is assigned. The list of extensions we obtain can be seen as a set of possible values that a record must take to be assigned a certain class. To assist the user during the use of the tool, a web interface has been also developed.

As future work, alternative techniques could be applied to break the symmetry of the graph so as to obtain causal relationship between arguments. Furthermore, particular attention could be paid to simplifying the explanation provided, including notions of symmetry and interchangeability between arguments, as well as applying Natural Language Processing to provide a further textual explanation. Finally, we could also apply Subjective Logic models [Jøsang, 2016] and use the weights on the BAF's edges to obtain a better explanation, instead of just using them for computing the probability of the extensions.

Riferimenti bibliografici

[Arrieta *et al.*, 2020] Alejandro Barredo Arrieta, Natalia Díaz Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, e Francisco Herrera. Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion*, 58:82–115, 2020.

[Baroni *et al.*, 2011] Pietro Baroni, Martin Caminada, e Massimiliano Giacomin. An introduction to argumentation semantics. *Knowl. Eng. Rev.*, 26(4):365–410, 2011.

[Bistarelli *et al.*, 2018] Stefano Bistarelli, Theofrastos Mantadelis, Francesco Santini, e Carlo Taticchi. Using metaproblog and conarg to compute probabilistic argumentation frameworks. In *AI*^{3}*@AI*IA*, volume 2296 of *CEUR Workshop Proceedings*, pages 6–10. CEUR-WS.org, 2018.

[Boella *et al.*, 2010] Guido Boella, Dov M. Gabbay, Leendert W. N. van der Torre, e Serena Villata. Support in abstract argumentation. In *COMMA*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 111–122. IOS Press, 2010.

[Borovcnik, 2012] Manfred Borovcnik. Conditional probability – a review of mathematical, philosophical, and educational perspectives. In *ICME, "Teaching and Learning Probability"*, 2012.

[Cayrol e Lagasquie-Schiex, 2005] Claudette Cayrol e Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *ECSQARU*, volume 3571 of *Lecture Notes in Computer Science*, pages 378–389. Springer, 2005.

[Cyras *et al.*, 2021] Kristijonas Cyras, Antonio Rago, Emanuele Albini, Pietro Baroni, e Francesca Toni. Argumentative XAI: A survey. In *IJCAI*, pages 4392–4399. ijcai.org, 2021.

[Dung, 1995] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[Fayyad e Irani, 1993] Usama M. Fayyad e Keki B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI*, pages 1022–1029. Morgan Kaufmann, 1993.

[Jøsang, 2016] Audun Jøsang. *Subjective Logic - A Formalism for Reasoning Under Uncertainty*. Artificial Intelligence: Foundations, Theory, and Algorithms. Springer, 2016.

[Kendall, 1974] William B. Kendall. A new algorithm for computing correlations. *IEEE Trans. Computers*, 23(1):88–90, 1974.

[Kulesza *et al.*, 2015] Todd Kulesza, Margaret M. Burnett, Weng-Keen Wong, e Simone Stumpf. Principles of explanatory debugging to personalize interactive machine learning. In *IUI*, pages 126–137. ACM, 2015.

[Li *et al.*, 2011] Hengfei Li, Nir Oren, e Timothy J. Norman. Probabilistic argumentation frameworks. In *TAFA*, volume 7132 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2011.

[von Eschenbach, 2021] Warren J. von Eschenbach. Transparency and the black box problem: Why we do not trust ai. *Philos Technol*, 34(4):1607–1622, 2021.