

An Interpretable Probabilistic Approach for Demystifying Black-box Predictive Models

Catarina Moreira*, Yu-Liang Chou, Mythreyi Velmurugan, Chun Ouyang, Renuka Sindhgatta, Peter Bruza

School of Information Systems, Queensland University of Technology, Brisbane, Australia

Abstract

The use of sophisticated machine learning models for critical decision making is faced with a challenge that these models are often applied as a ‘black-box’. This has led to an increased interest in interpretable machine learning, where *post hoc* interpretation presents a useful mechanism for generating interpretations of complex learning models. In this paper, we propose a novel approach underpinned by an extended framework of Bayesian networks for generating post hoc interpretations of a black-box predictive model. The framework supports extracting a Bayesian network as an approximation of the black-box model for a specific prediction. Compared to the existing post hoc interpretation methods, the contribution of our approach is three-fold. Firstly, the extracted Bayesian network, as a probabilistic graphical model, can provide interpretations about not only what input features, but also why these features contributed to a prediction. Secondly, for complex decision problems with many features, a Markov blanket can be generated from the extracted Bayesian network to provide interpretations with a focused view on those input features that directly contributed to a prediction. Thirdly, the extracted Bayesian network enables the identification of four different rules which can inform the decision-maker about the confidence level in a prediction, thus helping the decision-maker assess the reliability of predictions learned by a black-box model. We implemented the proposed approach, applied it in the context of two well-known public datasets and analysed the results, which are made available in an open-source repository.

Keywords: Interpretable machine learning, post hoc interpretation, probabilistic inference, Bayesian Network, predictive analytics

*Corresponding author

Email address: catarina.pintomoreira@qut.edu.au (Catarina Moreira)

1. Introduction

The rapidly growing adoption of Artificial intelligence (AI) has led to the development of supervised machine learning, in particular, deep neural networks, for generating predictions of high accuracy [1]. While the advancement has the potential to make significant improvement to the state-of-the-art in operational decision making across various business domains and processes, the underlying models are often opaque and do not provide the decision-maker with any understanding of their internal predictive mechanisms. This opaqueness in machine learning models is known as the *black-box* problem. Immediate consequences of trusting predictions from opaque models might result in severe losses for businesses (and people), unfair job losses, or even lead to negative impacts in certain societal groups (for instance, racial and gender discrimination) [2]. This has posed an open challenge to data scientists and business analysts on how to endow machine intelligence with capabilities to explain the underlying predictive mechanisms in a way that helps decision-makers understand and scrutinize the machine learned predictions.

The recent body of literature in machine learning has emphasised the need to interpret and explain the (machine) learned predictions. Methods and techniques have been proposed for explaining black-box models which are known as interpretable machine learning [3] or, in a broader context, explainable AI (XAI) [4]. So far, there exist two different mechanisms to address model interpretability. One is to have an interpretable model that provides transparency at three levels: the entire model, the individual components and the learning algorithm [5]. For example, both linear regression models and decision tree models are interpretable models. Another mechanism to address model interpretability is via *post hoc* interpretation, in which case, explanations and visualisations are extracted from a learned model, that is, after the model has been trained, and as such they are model agnostic. This is particularly useful for generating model interpretations for those complex machine learning models (such as deep neural networks) that have low transparency and are hard to be transformed into an interpretable model (i.e., a ‘white-box’) due to their sophisticated internal representations. The existing post hoc interpretation techniques (see a review in [3]) present knowledge about the various levels of impact of individual input features on the corresponding prediction.

In this paper, we propose a novel approach underpinned by an extended framework of Bayesian networks for generating post hoc interpretations of a black-box predictive model, with a focus on providing interpretations for any instance of prediction learned by the model (known as local interpretations). We name this framework the *Local Interpretation-Driven Abstract Bayesian Network* (LINDA-BN), which supports extracting a Bayesian network as an approximation (or an abstraction) of a black-box model for a specific prediction learned from any given input. We implemented our approach, applied it in the context of two well-known public datasets and analysed the results, which are made available in an open-source repository. Compared to the existing post hoc interpretation methods, the contribution of our approach is three-fold.

- The extracted Bayesian network not only can provide interpretations about *what* input features contributed to the corresponding prediction. As a probabilistic graphical model, it also represents knowledge about dependencies (in form of conditional probabilities) between input features and prediction, thus generating interpretations about *why* certain input features contributed to the prediction.
- For complex decision problems with a large number of features, the extracted Bayesian network is often complicated to be analysed by human. In this case, LINDA-BN supports generating a Markov blanket from the extracted Bayesian network. The Markov blanket determines the boundaries of a decision system in a statistical sense, and presents a graph structure covering a decision (e.g., a prediction), its parents, children, and the parents of the children. As such, the Markov blanket of the extracted Bayesian network provides interpretation with a focused view on those input features that directly contributed to the corresponding prediction.
- The extracted Bayesian network enables the identification of four different rules which can inform the decision-maker about the confidence level in a given prediction. As such, the interpretations provided in our approach can help the decision-maker assess the reliability of predictions learned by a black-box model.

In the rest of the paper, we continue to introduce the relevant concepts and review the related research efforts in Section 2. We present our approach underpinned by the framework

LINDA-BN in Section 3. Next, we report the experiments and discuss the results of analysis in Section 4. Finally, we conclude the paper with an outlook to future work (Section 5).

2. Background and Related Work

In this section, we present the main concepts that are used throughout our work, and review research efforts that are related to the proposed framework.

2.1. Concepts

Prior to discussing existing work that relates to our approach on providing interpretations of a *black box* machine learning model prediction, we note the following definitions:

- **Black box predictor:** It is a machine learning opaque model, whose internals are either unknown to the observer or they are known but are not understandable by humans.
- **Interpretability:** The ability to extract symbolic information out of a black box that can provide the meaning in understandable terms to a human [6].
- **Explainability:** The ability to highlight decision-relevant parts of the used representations of the algorithms and active parts in the algorithmic model, that either contribute to the model accuracy on the training set, or a specific prediction for one particular observation [7].

One can see interpretability as the extraction of symbolic information from the black box (machine-level) that already needs some degree of semantics, and explainability as the conversion of this symbolic information to a human understandable way (human-level).

2.2. Related Work

Various approaches have been proposed in the literature to address the problem of interpretability. Generally, this problem can be classified into two major models: Interpretable models and model agnostic (post-hoc) models.

Interpretable models are by design already interpretable, providing the decision-maker a transparent white box approach for prediction. Decision tree, logistic regression, and linear

regression are commonly used interpretable models. These models have been used to explain predictions of specific prediction problems [8]. Model-agnostic approaches, on the other hand, refer to the deriving explanations from a black box predictor by extracting information about the underlying mechanisms of the system. In addition, studies have focused on providing model-specific post-hoc explanations [9]. The focus of our work is to build model-agnostic post-hoc methods as they have flexibility of being applied to any predictive model as compared to model-specific post-hoc approaches . To discover the demystifying predictive black box models, we focus on the widely cited post-hoc models that include LIME [10], SHAP [11], and Counterfactual explanation in this work.

2.2.1. *LIME*

Local Interpretable Model-agnostic Explanations (LIME) [10] explains the predictions of any classifier by approximating it with an locally faithful interpretable model. Hence, LIME generates local interpretations by perturbing a sample around the input vector within a local decision boundary [12, 10]. Each feature is associated with a weight that is computed using a similarity function that measures the distances between the original instance prediction and the predictions of the sampled points in the local decision boundary. Linear regression is learned to determine the local importance of each feature.

LIME has been extensively applied in the literature. For instance, Stiffler et al. [13] used LIME to generate salience maps of a certain region showing which parts of the image affect how the black box model reaches a classification for a given test image. Tan et al. [14] apply LIME to demonstrate the presence of uncertainty in the explanations that could raise concerns in the use of the black box model and diminish the value of the explanations. different sources of uncertainty in the explanation. Their work demonstrates the presence of three sources of uncertainty: randomness in the sampling procedure, variation with sampling proximity, and variation in explained model across different data points. Anchor [15] is an extension of LIME that attempts to address some of the limitations by maximizing likelihood on how a certain feature might contribute to a prediction. Anchor introduces IF-THEN rules as explanations as well as the notion of coverage, which allows the decision-maker to understand the boundaries in which the generated explanations are valid.

2.2.2. SHAP

The SHAP (SHapley Additive exPlanations) is an explanation method which uses Shapley values [16] from coalitional game theory to fairly distribute the gain among players, where contributions of players are unequal [11]. Shapely values are a concept in economics and game theory and consist in a method to fairly distribute the payout of a game among a set of players. One can map these game theoretic concepts directly to an XAI approach: a game is the prediction task for a single instance; the players are the feature values of the instance that collaborate to receive the gain. This gain consists of the difference between the Shapley value of the prediction and the average of the Shapley values of the predictions among the feature values of the instance to be explained [17].

Strumbelj and Kononenko [17] claim that in a coalition game, it is usually assumed that n players form a grand coalition that has a certain value. Given that we know how much each smaller (subset) coalition would have been worth, the goal is to distribute the value of the grand coalition among players fairly (that is, each player should receive a fair share, taking into account all sub-coalitions). Lundberg and Lee [11] on the other hand, present an explanation using SHAP values and the differences between them to estimate the gains of each feature.

In order to fairly distribute the payoff amongst players in a collaborative game, SHAP makes use of two fairness properties: (1) Additivity, which states that amounts must sum up to the final game result, and (2) Consistency, which states that if one player contributes more to the game, (s)he cannot get less reward.

In terms of related literature, Miller Janny Ariza-Garzón and Segovia-Vargas [18] adopted SHAP values to assess logistic regression model and several machine learning algorithms for granting scoring in P2P (peer-to-peer) lending, the authors point out SHAP values can reflect dispersion, non-linearity and structural breaks in the relationships between each feature and the target variable. They concluded that the SHAP can provide accurate and transparent results on the credit scoring model. Parsa et al. [19] also highlight that SHAP could bring insightful meanings to interpret prediction outcomes. For instance, one of the techniques in the model, XGBoost, not only is capable of evaluating the global importance of the impacts of features on the output of a model, but it can also extract complex and non-linear joint

impacts of local features.

2.2.3. Probabilistic graphical model

The literature of interpretable methods for explainable AI based on probabilistic graphical models (PGM) is mostly dominated by models based on counterfactual reasoning in order to derive explanations for a specific local datapoint.

The counterfactual explanation based on PGM comprises of a conditional assertion whose antecedent is false and whose consequent describes how the world would have been if the antecedent had occurred. It provides interpretations as a mean to point out which changes would be necessary to accomplish the desired goal, rather than supporting the understanding of why the current situation had a certain predictive outcome [20]. For instance, in a scenario where a machine learning algorithm assesses whether a person should be granted a loan or not, a counterfactual explanation of *why* a person did not have a loan granted could be in a form of a scenario *if your income was greater than \$15,000 you would be granted a loan* [21]. Unlike other explanation methods that depend on approximating an interpretable model within a perturbed decision boundary, counterfactual explanations have the strength that it is always truthful to the underlying model by providing direct outputs of the algorithms [10].

Counterfactual explanations are part of causal inference methods, which are based on causal reasoning. and is focused on the estimation of the causal effects from treatments and actions [22]. In 2000, Pearl proposed a framework (the ladder of causation) that proposes different levels of causal relationships during causal inference. Level 1, *Association*, entails the sensing of regularities or patterns in the input data, expressed as relations; it focuses on the question *what*. Level 2, *Intervention*, predicts the effects of deliberate actions, expressed as causal relationships. And Level 3, *Counterfactuals*, involve constructing a theory of the world that explains why certain actions have specific effects and what happens is the absence of such actions [22]. A simple and naive approach for generating counterfactual explanations is searching by trial and error. In this approach the feature values are randomly changed for the instance of interest and stops searching when the desired output is predicted.

The notion of counterfactual model has been investigated by a few researchers. In 2013, the counterfactual approach has been proposed for the evolution of advertisement place-

ment in search engines [23]. Johansson et al. [24] claim that the counterfactual thinking has been adopted in the context of machine learning applications to predict the result of several different actions, policies, and interventions using non-experiment data. Moreover, the Counterfactual Gaussian Process (CGP) approach has been created by Schulam and Saria [25] for modelling the effects of sequences of actions on continuous time series data and facilitate the reliability of medical decisions [26].

Although counterfactual explanation are useful, they do not explain why a certain prediction is made. On the contrary, they assume a hypothetical scenario where the prediction would be contrary to the output of that particular data point. Our approach aims to use probabilistic model to provide local explanations that provide insights into the features influencing a datapoint.

3. The Local Interpretation-Driven Abstract Bayesian Network Framework

In this section, we present our framework built upon an extended framework of Bayesian networks that can generate post hoc interpretations for a single data point of prediction: the local interpretation-driven abstract Bayesian network (LINDA). We start with a brief introduction to Bayesian networks (Section 3.1) and structure learning (Section 3.2). Readers that are familiar with the knowledge can proceed directly to the proposed framework (Sections 3.3 to 3.5).

3.1. Bayesian Networks

A Bayesian Network (BN) is a directed acyclic graph in which each node represents a random variable, and each edge represents a direct influence from the source node to the target node. The graph represents (in)dependence relationships between variables, and each node is associated with a conditional probability table that specifies a distribution over the values of the node given each possible joint assignment of the values of its parents [27].

Bayesian networks can represent essentially any full joint probability distribution, which can be computed using the chain rule in probability theory [28]. Let \mathcal{G} be a BN graph over the variables X_1, \dots, X_n . We say that a probability distribution, Pr , over the same space

factorizes according to \mathcal{G} , if Pr can be expressed using the following equation [29]:

$$Pr(X_1, \dots, X_n) = \prod_{i=1}^n Pr(X_i | Pa_{X_i}). \quad (1)$$

In Equation 1, Pa_{X_i} corresponds to all the parent variables of X_i . The graph structure of the network, together with the associated factorization of the joint distribution allows the probability distribution to be used effectively for inference (i.e. answering queries using the distribution as our model of the world). For some query Y and some observed variable e , the exact inference in Bayesian networks is given by the following equation [29]:

$$Pr(Y|E = e) = \alpha Pr(Y, e) = \alpha \sum_{w \in W} Pr(Y, e, w), \quad \text{with } \alpha = \frac{1}{\sum_{y \in Y} Pr(y, e)}. \quad (2)$$

Each instantiation of the expression $Pr(Y = y, e)$ can be computed by summing up all joint entries that correspond to assignments consistent with y and the evidence variable e . The set of random variables W corresponds to variables that are neither query nor evidence. The α parameter specifies the normalization factor for distribution $Pr(Y, e)$, and this normalization factor is informed by certain assumptions made in Bayes rule [28].

3.2. Structure Learning in Bayesian Networks

Bayesian networks are made of two important components: a directed acyclic graph \mathcal{G} representing the network structure, and a set of probability parameters Θ representing the conditional dependence relations. Learning a BN is a challenging problem when the network representation \mathcal{G} is unknown. Given a dataset \mathcal{D} with m observations, $Pr(\mathcal{G}, \Theta|\mathcal{D})$ is composed of two steps, structure learning and parameter learning, as follows [30]:

$$Pr(\mathcal{G}, \Theta|\mathcal{D}) = \underbrace{Pr(\mathcal{G}|\mathcal{D})}_{\text{structure learning}} \cdot \underbrace{Pr(\Theta|\mathcal{G}, \mathcal{D})}_{\text{parameter learning}}. \quad (3)$$

Structure learning aims to find the directed acyclic graph \mathcal{G} by maximising $Pr(\mathcal{G}|\mathcal{D})$. Parameter learning, on the other hand, focuses on estimation of the parameters Θ given the graph \mathcal{G} obtained from structure learning. According to [31, 32], considering that parameters Θ represent independent distributions (as assumed in Naïve Bayes), the learning process can

be formalised as follows [30]:

$$Pr(\Theta|\mathcal{G}, \mathcal{D}) = \prod_i Pr(\Theta_{X_i}|\Pi_{X_i}, \mathcal{D}). \quad (4)$$

It is important to note that structure learning is well known to be both NP-hard [33] and NP-complete [34] due to the following equation:

$$Pr(\mathcal{G}|\mathcal{D}) \propto Pr(\mathcal{G})Pr(\mathcal{D}|\mathcal{G}), \quad (5)$$

which can be decomposed into:

$$\begin{aligned} Pr(\mathcal{D}|\mathcal{G}) &= \int Pr(\mathcal{D}|\mathcal{G}, \Theta)Pr(\Theta|\mathcal{G})d\Theta \\ &= \prod_i \int Pr(X_i|\Pi_{X_i}, \Theta_{X_i})Pr(\Theta_{X_i}|\Pi_{X_i})d\Theta_{X_i} \end{aligned} \quad (6)$$

In structure learning, it is often used the BIC score, a frequentist measure, to maximise, $Pr(\mathcal{G}, \Theta|\mathcal{D})$, due to its simplicity.

$$Score(\mathcal{G}, \mathcal{D}) = BIC(\mathcal{G}, \theta|\mathcal{D}) = \sum_i logPr(X_i|\Pi_{X_i}, \Theta_{X_i}) - \frac{\log(n)}{2} |\Theta_{X_i}|. \quad (7)$$

According to Scutari et al. [30], structure learning via score maximisation is performed using general-purpose optimisation techniques, typically heuristics, adapted to take advantage of these properties to increase the speed of structure learning. The most common are greedy search strategies that employ local moves designed to affect only few local distributions, so that new candidate DAGs can be scored without recomputing the full $Pr(\mathcal{D}|\mathcal{G})$. This can be done either in the space of the DAGs with hill climbing and tabu search [28]. In this paper, we opted for a greedy Hill Climbing approach to learn the structure \mathcal{G} , due to its simplicity and effective results [31].

3.3. Local Interpretation-Driven Abstract Bayesian Network (LINDA-BN)

State-of-the-art techniques for constructing predictive models underpinned by machine intelligence usually adopt a ‘black-box’ approach, where the reasoning behind the predictions remains opaque (particularly in regard to deep learning models). Consequently, the underlying predictive mechanisms remain largely incomprehensible to the decision-maker.

The challenge is how to endow machine intelligence with capabilities to explain the underlying predictive mechanisms in a way that helps decision-makers understand and scrutinize the machine learned decisions. In the following, we propose an extended framework of Bayesian Networks for generating post hoc local interpretations of black-box predictive models. We name this framework the *Local Interpretation-Driven Abstract Bayesian Network* (LINDA-BN). It supports extracting a Bayesian network as an approximation (or an abstraction) of a black-box model for a specific prediction learned from any given input. Note that explanations can be constructed from the graphical representations of LINDA-BN, and we will address the explanation generation component as a direction for future work.

The basic idea behind the proposed framework LINDA-BN rests in three main steps: i) permutation generation, ii) Bayesian network learning, and iii) computation of the Markov Blanket of the class variable (representing result of a prediction). It is important to stress that the proposed model aims to augment a decision-maker's intelligence towards a specific decision problem, providing interpretations that can either reinforce the predictions of the black-box or lead to a complete distrust in these predictions (identification of misclassifications). Figure 1 shows a general illustration of the proposed framework.

Given a vector of input features $\vec{X} = \{x_1, x_2, \dots, x_n\}$ and a black-box predictor, $\hat{y}(\vec{X})$, the goal is to introduce a set of permutations \vec{X}'_i in the features of \vec{X} in a permutation variance $\epsilon \in [0, 1]$ in such a way that each feature will be permuted using a uniform distribution over the interval $[x_i - \epsilon, x_i + \epsilon]$. The goal is to analyse how introducing a small perturbation can impact the predictions of the black box prediction, $\hat{y}(\vec{X}'_i)$, generating a new statistical distribution describing small variations of the input vector \vec{X} . The goal is to learn a Bayesian network structure out of this statistical sample using a Greedy Hill Climbing approach. Our hypothesis is the following: if the data point falls within the correct decision region of the black box predictor, leading to a correct class classification, c , then the predictions of all the permutations, $\hat{y}(\vec{X}'_i)$, should be close to certainty, i.e. favouring one of the assignments of the class variable with $Pr(Class = c | X'_i) \approx 1$. This can strengthen the decision-maker's trust in the predictions of the black-box predictor. If, however, the data point, \vec{X} , is very close to the black-box's decision boundary, then one would expect that the permutations will be spread around the different regions demarcated by the decision boundary, leading

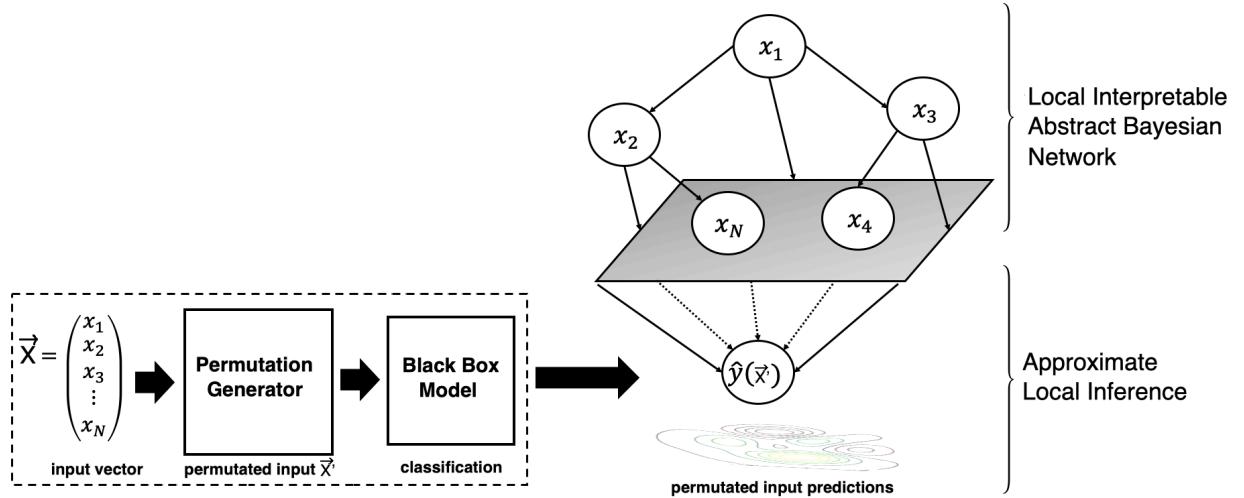


Figure 1: An general illustration of the proposed framework LINDA-BN

to a more diversified statistical distributions of predictions, and a higher uncertainty in the classification of the respective class $Pr(Class = c | X'_i) \ll 1$. Such situations have the potential to alert the decision-maker that the black-box predictor is not very certain about the classification of the given data point. Section 3.5 is centered in this topic.

Since the network structure shows dependencies between the input features and the class variable, then it is possible to extract what features contributed to the prediction and *why*, allowing a deeper understanding about the impact that the features have in the class variable, or even provide the decision-maker additional insights about the decision problem.

For complex decision problems with a large amount of features, the local interpretable network that is learned from the generated permutations is extremely complicated to be analysed by a human, so a Markov Blanket is returned, instead, as a summarisation of what are the main variables influencing the class variable. The Markov blanket determines the

Algorithm 1: Local Interpretation-Driven Abstract Bayesian Network Generator

Input: *local_vec*, single vector from which we want to generate interpretations
 black_box, a predictive model
 ϵ , variance range to permute the features (default = 0.1)
 n_samples, number of permuted samples to generate (default = 300)
 class_var, string with the name of the class variable
Output: \mathcal{G} , the Local Interpretable Abstract Bayesian Network

```
1: /* Generate permutations via a uniform distribution within a permutation range */  
2: perms = GeneratePermutations( x, model,  $\epsilon$ , n_samples )  
3:  
4: /* Discretise continuous features according to the number of quartiles */  
5: perms_discr = DiscretisePermutations( perms, quartiles = 4 )  
6:  
7: /* Learn BN from discrete permutations using a Greedy Hill Climbing Search */  
8: bn = LearnBN_GreedyHillClimbing( perms_discr )  
9:  
10: /* Compute BN's marginal distributions */  
11: bn_inf = ComputeMarginalDistributions( bn )  
12:  
13: /* Compute BN's Markov blanket */  
14: bn_markov = ComputeMarkovBlanket( bn, class_var )  
15:  
16: if bn.nodes <= 10 then  
17:   return bn_inf /* return full network */  
18: else  
19:   return bn_markov /* return Markov blanket */  
20: end if  
21:
```

boundaries of a system in a statistical sense. It includes all its parents, children, and the parents of the children.

It can be shown that a node is conditionally independent of all other nodes given values for the nodes in its Markov blanket. Hence, if a node is absent from the class attribute's Markov blanket, its value is completely irrelevant to the classification [29]. Algorithm 1 describes the algorithm that we used for to generate the proposed local interpretable abstract Bayesian network.

3.4. Interpreting Graphical Representations through Reasoning

This section analyses how to interpret the different situations where a random variable can influence another in the local interpretable Bayesian network model.

In a common cause structure, Figure 2 (a), the local interpretable model approximates to a Naïve Bayes classifier, which means that having knowledge about the class variable will make the feature variables X_1, X_2, \dots, X_N conditionally independent, and consequently uncorrelated. This means that knowing about X_1 does not bring any additional information to the decision-maker. Although human decision-makers tend to assess and interpret these structures as cause/effect relationships as a way to simplify and linearise the decision problem due to bounded rationality constraints, statistically, common cause structures do not imply causal effects in Bayesian networks [27]. The consideration of the class variable as a prior in interpretations for a *single datapoint* may indicate a high uncertainty obtained in the statistical sample of the permuted features, suggesting that the datapoint that is being interpreted may be very close to the predictive black-box decision boundary (Section 3.5 addresses this with a higher detail).

The other type of structure that one can often find in the local interpretable model is the *v*-structure, also called common effect (Figure 2 (b), which approximates to a linear regression representation. This means that, the features become conditionally independent of the class, if and only if one has knowledge about the class variable. Being uncertain about the class will lead to an influence from the features, X_1, X_2, \dots, X_N . In terms of the proposed local interpretable model, this means that the features have a direct effect in the class variable, and humans can interpret it through an abductive reasoning process.

Abduction is a mode of human reasoning, which was brought into prominence by the American philosopher C.S. Peirce [35]. In Peirce’s view abduction is an inference of the form: “The surprising fact C is observed. But if A were true, C would be a matter of course. Hence, there is reason to suspect that A is true”. Abductive inference is thus a process of justifying an assumption, hypothesis or conjecture in producing the class of interest. Peirce states that abduction might explain a given set of data, or might facilitate observationally valid predictions, or might permit the discounting of other hypotheses. By engaging in abduction, the decision maker interpreting the graph structure is afforded a *simpler* and *more compact*

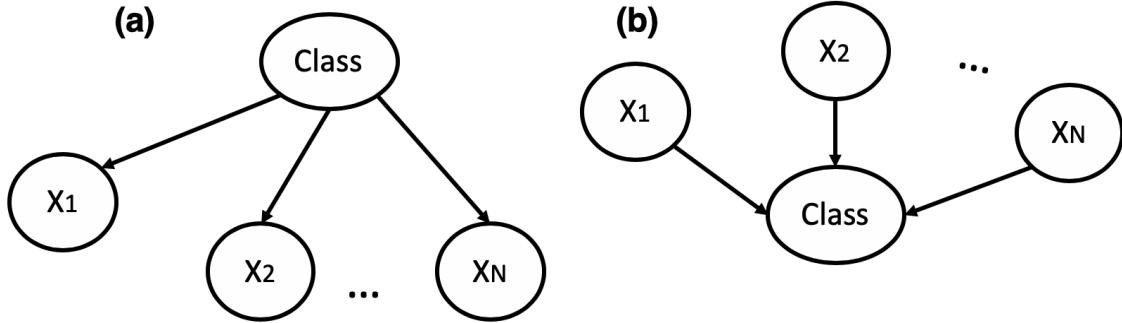


Figure 2: Different graph structures for probabilistic reasoning

account [35].

Abduction is not a sound form of inference like deduction, and so even though the decision maker might suspect A , there is a degree of uncertainty. Abduction is sometimes termed “inference to the best explanation” where there is no guaranteed certainty in the explanation. In other words, given a set of observations, the decision maker uses abduction to find the simplest, most likely and compact explanation from the graph structure. The Markov blanket of the class variable is a way of supporting the decision maker’s abductive reasoning process.

3.5. Rules for Local Interpretations

The graphical nature of the proposed framework LINDA-BN enables the identification of certain parts that can help the decision-maker assess the reliability of the predictions of the black box for single datapoints. To this end, we propose a set of four rules that correspond to four different patterns that the proposed model can identify, depending on how close to the decision boundary a datapoint is. By analysing the confidence of the interpretable model

with regards to the class variable together with the structure of the network, one can provide useful guidelines to the decision-maker that can be later be used to generate human centric and understandable explanations (which is not the focus of this work).

The proposed rules to assess the confidence of the black box predictions using the proposed framework are the following:

- **Rule 1: High confidence in predictions.** *If the black box predicts a class c for a given datapoint, \vec{X} , and the class variable is contained in a common-effect structure in \mathcal{G} with a probability $Pr(\text{Class} = c) \approx 1$, then the interpretable model, \mathcal{G} , supports the prediction of \vec{X} and its respective Markov blanket determines the most relevant features.*

As mentioned in Section 3.4, common-effect structures in \mathcal{G} approximate to a linear regression representation in which there is a direct influence from the features to the class. When $Pr(\text{Class} = c) \approx 1$, then this means that the datapoint falls in a well defined decision region, as illustrated in Figure 3. Since the likelihood of the class is close to certainty, the decision-maker can make use of the class' respective Markov blanket for explanation and perform an abductive reasoning process in which the decision-maker will seek to find the simplest and most likely conclusion out of the Markov blanket.

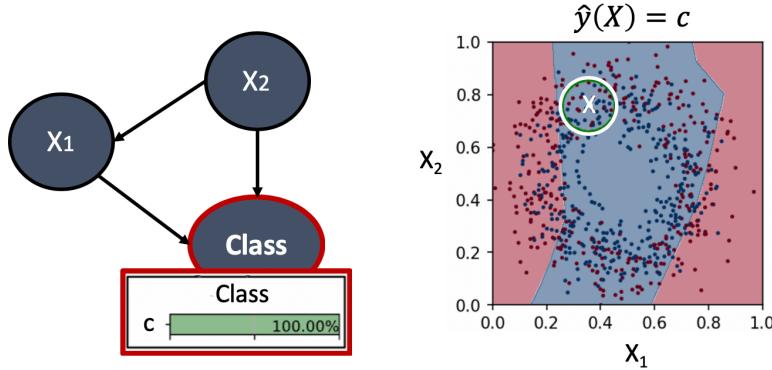


Figure 3: Graphical representation of a pattern representing Rule 1, a high confidence in the prediction of the black box, supported by an interpretable graph showing what are the most relevant features influencing the class variable.

- **Rule 2: Unreliable predictions.** *If the interpretable network, \mathcal{G} , has a structure where the class variable is independent from all other feature variables, that is $\text{Class} \perp \{X_1, \dots, X_N\}$, then this corresponds to an unrealistic decision scenario, because the features are uncorrelated from the class variable and providing information*

about them does not make any change in the probability $\Pr(\text{Class} = c)$. Thus, the classification $\hat{y}(\vec{X})$ is incorrect and it should be communicated to the decision-maker as an unreliable prediction.

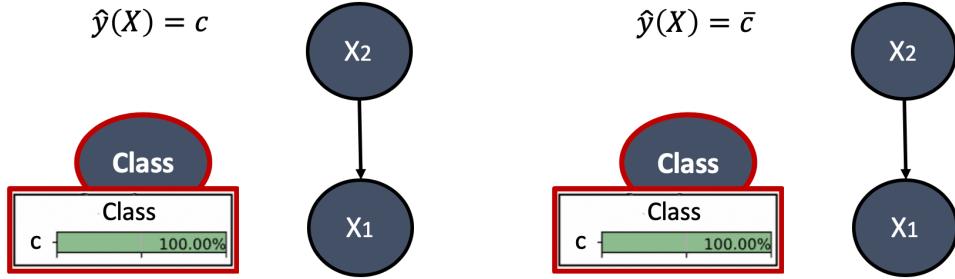


Figure 4: Graphical representation of a pattern representing Rule 2, a distrusted prediction of the black box, supported by an interpretable graph showing that knowing information about the features does not make any changes in the class variable.

Sometimes, due to problems in generalising the black box predictor, there can be classifications that are erroneous and unrealistic. In these rare scenarios, the Local Interpretable model can learn from the permuted instances of \vec{X} , a graphical structure in which $\text{Class} \perp \{X_1, \dots, X_N\}$ (Figure 4 shows an example). In these situations, the Markov Blanket contains only the class variable, which makes it easy to identify the independence in the class variable. Moreover, it can be easily concluded that the classification $\hat{y}(\vec{X})$ is incorrect and it should be communicated to the decision-maker as an unreliable and unrealistic prediction that results from poor generalisation of the black box.

- **Rule 3: Contrast Effects.** *If the black box predicts $\hat{y}(\vec{X}) = c$, and the maximum likelihood of the class variable in \mathcal{G} is $\Pr(\text{Class} = \bar{c})$, then there is a contradiction between the local interpretable abstract model and the prediction computed by the black box, suggesting that the datapoint is very close to the decision boundary, which can either be correctly or incorrectly classified. Thus, the decision-maker should analyse the Markov Blanket of the class variable representing \vec{X} , and assess whether the relationships between the features justify the class.*

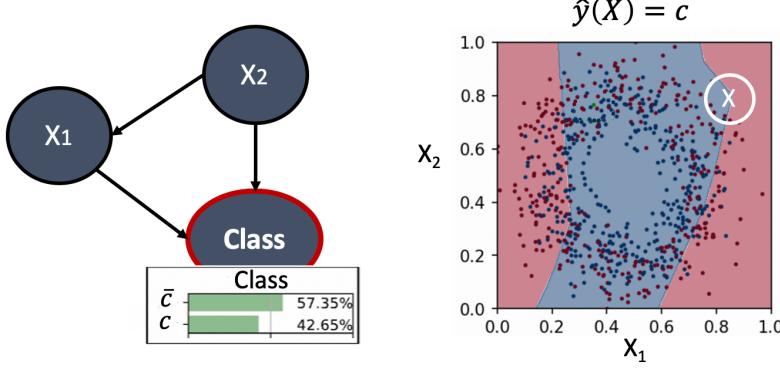


Figure 5: Graphical representation of a pattern representing Rule 3, a contrast effect, where the local interpretable abstract model reinforces a class that is different from the one predicted by the black-box.

In situations where the datapoint is very close to a decision boundary, the permutation of the datapoint \vec{X} will generate a statistical distribution within a certain neighbourhood of X . Due to the complexity and non-linearity of the decision boundary, the statistical distribution can increase the likelihood, $Pr(Class = \bar{c})$, contradicting the prediction of the black box, $\hat{y}(\vec{X}) = c$. In these situations, even if the black-box managed to predict correctly X , there is a high uncertainty in the prediction, and it should be recommended to the decision-maker to assess the features of X in order to assess its reliability. Figure 5 shows an example of a contrast effect.

- **Rule 4: Uncertainty in predictions.** *If the black box predicts $\hat{y}(\vec{X}) = c$, and the probability of the class variable in \mathcal{G} is $Pr(Class = c) \ll 1$, but still with a maximum likelihood favouring class c , then datapoint X falls near the decision boundary. Even if the class is in accordance with the prediction of the black-box, then there is an underlying uncertainty attached to its prediction. Thus, the decision-maker should analyse the Markov Blanket of the class variable representing \vec{X} , and assess whether the relationships between the features justify the class.*

This situation is very similar to the contrast effect (rule 3) with the difference that the class variable in \mathcal{G} is still consistent with the predictions of $\hat{y}(\vec{X})$. However, the statistical distribution of the predictions of the permutations of \vec{X} have a high uncertainty and do not allow the decision-maker to be fully confident in the prediction of \hat{X} . Thus, depending on the degree of uncertainty of $Pr(Class = c)$, the decision-maker should

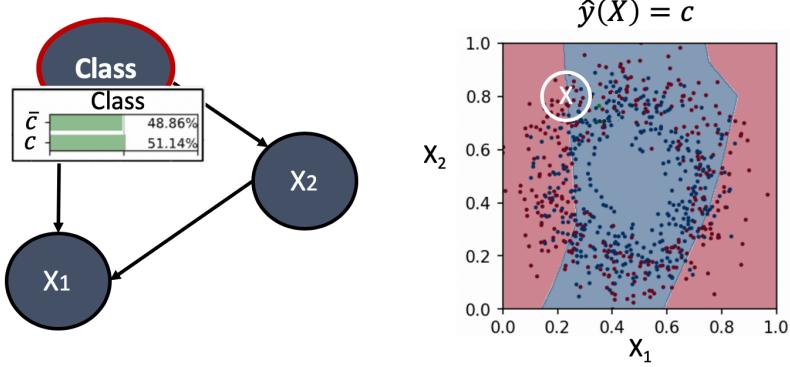


Figure 6: Graphical representation of a pattern representing Rule 4, uncertainty in the prediction, where the local interpretable abstract model shows that the black box prediction is as good as flipping a coin.

analyse the Markov Blanket of the class variable representing \vec{X} , and assess whether the relationships between the features justify the class. Figure 6 shows an example of an uncertain prediction. Although the likelihood of the variable *Class* is in accordance with $\hat{\vec{X}} = c$, the local interpretable abstract model shows full uncertainty in the prediction: the prediction is as good as flipping a coin.

4. Evaluation

Given that there are no standard evaluation metrics for XAI [3], in this , we present a thorough analysis of the proposed LINDA-BN model in accordance to the rules that we put forward in Section 3.5. We performed an analysis in terms of two public well-known datasets from the literature, namely the *Pima Indians diabetes dataset* and the Breast Cancer Wisconsin [36], both from the *UCI Machine Learning Repository*¹. We have made available a public repository with Jupyter notebooks with the proposed model and all the experiments that we made for this research work: https://github.com/catarina-moreira/LINDA_DSS.

In Section 4.1, we present the main experimental setup for our analysis. Section 4.2, presents an analysis of the impact of the permutation variance in the proposed LINDA model. In Section 4.3, we make a statistical analysis of the distribution of the interpretations generated by LINDA over both datasets and their the different rules together with existing interpretable approaches such as LIME [10] and SHAP [11]. Finally, Section 4.4, describes

¹<https://archive.ics.uci.edu/ml/index.php>

how the proposed interpretable model performs in more complex decision scenarios.

4.1. Design of Experiments

In order to assess the performance and interpretations generated by the proposed LINDA model, we trained a deep learning neural network for two two public well-known datasets from the literature, namely the *Pima Indians diabetes dataset* and the *Breast Cancer Wisconsin* datasets. Both datasets are highly unbalanced, and for that reason, we had to balance the datasets in order to not have a biased predictive model.

We performed a grid search approach in order to find the best performing neural network model. The characteristics of the models can be found in Table 1. As such, the learned models apply sophisticated internal working mechanisms and run as a black-box when making predictions.

Parameters	Diabetes	Breast Cancer
Model Accuracy	0.7380	0.9840
Num. Hidden Layers	5	4
Num. Neurons per Hidden Layer	12	12
Hidden Layer Activation function	Relu	Relu
Ouput Layer Activation function	Softmax	Softmax

Table 1: Deep neural network architecture found for the best performing model in the Diabetes and the Breast Cancer datasets.

4.2. Analysis of the Impact of Different Permutation Variances

As in other representative interpretable models in the literature (like LIME and SHAP), LINDA-BN performs permutations between an interval in the range $[x_i - \epsilon, x_i + \epsilon]$ on the input vector's features $\vec{X} = \{x_1, x_2, \dots, x_N\}$ in order to generate a statistical distribution of how the predictions of the black-box change with the features. To investigate the impact of the permutation variance, ϵ , we performed a set of experiments for both datasets, where we varied $\epsilon \in [0, 1]$, and analysed how many times the proposed interpretable model returned a structure that is consistent with the rules proposed in Section 3.5. The obtained results are summarised in Figure 7.

Taking a close look at the Diabetes dataset in Figure 7, results show that low variance makes the interpretable model very confident in its interpretations, with 92% of the datapoints (both from the training set and test set) falling in rule 1 (high confidence in the

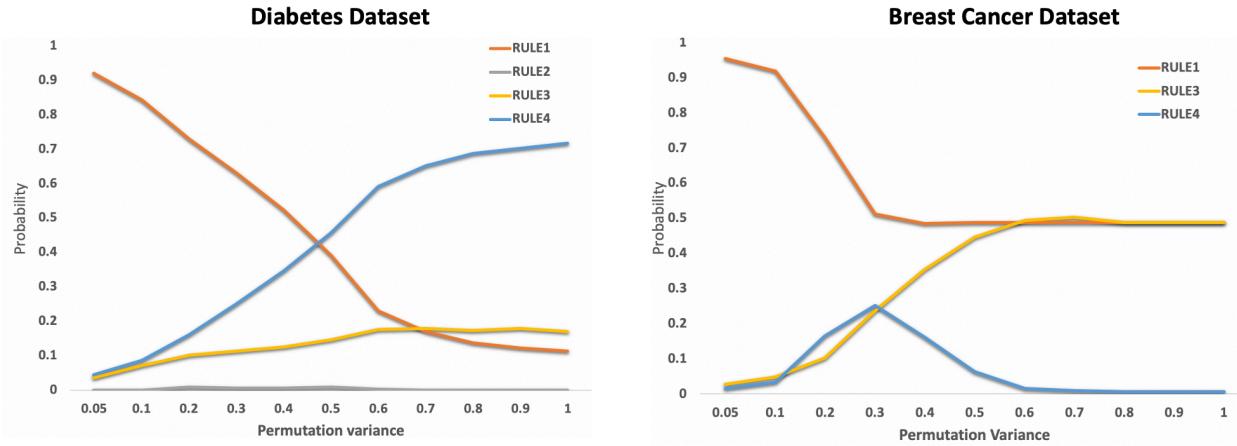


Figure 7: Impact of the permutation variance boundary in the different proposed rules.

prediction). This is due to the fact that, for a very small permutation interval, the probability of hitting a decision boundary is very small. This is confirmed with the verification of the low amount of datapoints falling in rule 4 (uncertainty in the predictions) or rule 3 (contrast effects). When the permutation variance starts to increase, the model becomes less certain about the predictions. Consequently, rule 1 starts to decrease exponentially, while rule 4 starts to show a lot in uncertainty in the interpretations generated for the different datapoints. One can see that when the permutation variance reaches half of the feature space (note that we assume that the features of the black-box are scaled between 0 and 1 as in standard machine learning applications), then there comes a point where the uncertainty is so high that the interpretable model stops having confidence in almost 90% of its interpretations. Additionally, almost 80% of the datapoints start falling in rule 4.

In terms of the impact of the permutation variance for the breast cancer dataset, the

scenario tends to be slightly different. Just like in the diabetes dataset, when the permutation variance interval is very small, the statistical distribution of the predictions learned by the proposed LINDA model majorly falls in rule 1. However, when the permutation variance starts to increase together with the uncertainty levels, we start to notice an increase of contrast effects (rule 3), rather than uncertainty in the predictions (rule 4). The reason for this is due to the effectiveness of the black-box. Note that the accuracy of the deep neural network model for the diabetes dataset was 73.80%, while the learned model for the breast cancer dataset achieved an accuracy of 98.40%. Since the majority of the datapoints fall in well defined decision regions, when the permutation variance increases, the statistical distribution of predictions will tend to show misclassifications (a statistical distribution more concentrated in the opposite region of the decision boundary). Figure 7 also shows that permutation variances superior to 0.2 do not decrease the certainty of the interpretability of correctly predicted datapoints, however it does increase the number of contrast effects (rule 3), reinforcing again the idea that bigger permutation intervals will make the distributions point towards opposite directions of the decision boundary.

Rules	Set	DIABETES				BREAST CANCER			
		TP	TN	FP	FN	TP	TN	FP	FN
Rule 1	Train	0.8662	0.91	0.7627	0.7419	0.9931	0.8786	1.0000	0.1667
	Test	0.7576	0.96	0.57	0.8571	1.0000	0.8286	1.0000	0.0000
Rule 2	Train	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	Test	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Rule 3	Train	0.0828	0.016	0.1186	0.0645	0.0000	0.0643	0.0000	0.6667
	Test	0.1818	0.0000	0.14	0.0000	0.0000	0.1143	0.0000	0.0000
Rule 4	Train	0.0509	0.0787	0.1186	0.1935	0.0069	0.0571	0.0000	0.1667
	Test	0.0606	0.04	0.29	0.1429	0.0000	0.0571	0.0000	0.0000

Table 2: Overview of the distribution of the datapoints for the Diabetes and Breast Cancer datasets over the proposed rules using LINDA in order to determine the confidence of the computed interpretations.

In order to extract interpretable models that are both highly confident in the interpretations, but can also flag possible misclassifications, we decided to set the permutation variance to 0.1 for the remaining parts of this analysis.

4.3. Analysis of Rules for Local Interpretations

In this section, we analyse the impact of the proposed rules in the different classifications: true positives, true negatives, false positives, and false negatives. The goal with this analysis is to understand if the proposed rules can provide the decision-maker some insights on whether or not, the decision-maker is facing a correct classification or a possible misclassification. Table 2 shows the percentage of datapoints over the different proposed rules for both the diabetes and breast cancer datasets, using $\epsilon = 0.1$.

- **Correct classifications majorly coincide with rule 1, leading to highly confident predictions.** For the breast cancer dataset, for instance, all datapoints classified as true positives and true negatives were categorised as rule 1 with high confident interpretations. For the diabetes dataset, since the black box had a more poor performance, then the percentage of correctly classified datapoints is already smaller. Still, 86% of the true positives in the training set fell under the category of rule 1 and in the test set 75%. Regarding the true negatives, more than 90% of the datapoints had interpretations supporting a true classification of a non-diabetes case. When compared with interpretations from state of the art algorithms, one can see that both LIME and SHAP also tend to reinforce the features that are contributing positively to the class diabetes. Figure 8 shows an example of an interpretation of a correctly classified datapoint ($\epsilon = 0.1$) and the respective interpretations for LIME and SHAP. For the case of misclassifications, there was a significant amount of datapoints belonging to the set of the false positives and the false negatives that were also categorised as rule 1. In these examples, the interpretable model could not provide significant insights of why there was a misclassification.

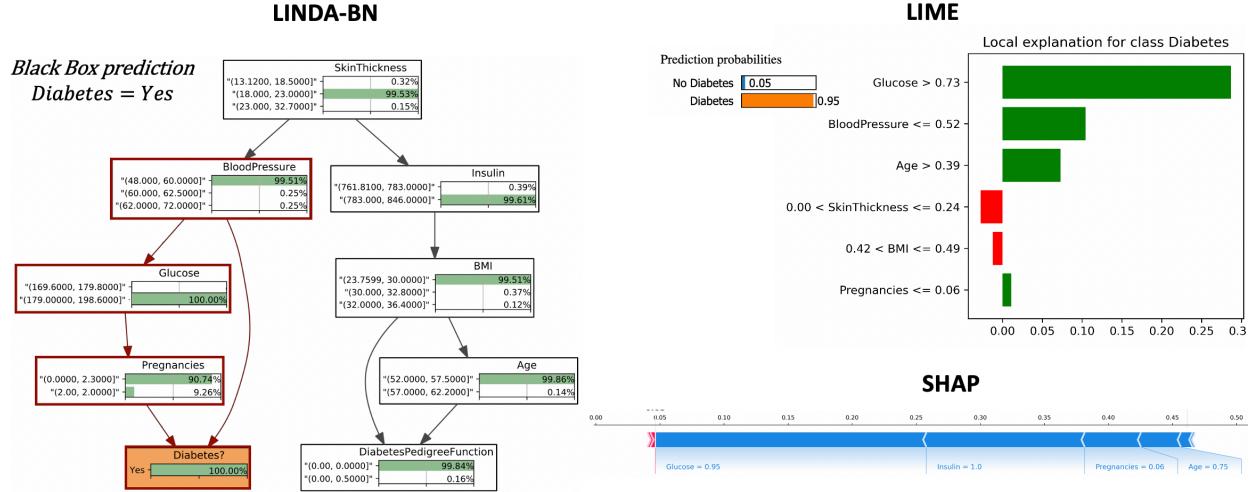


Figure 8: Correct prediction high confidence (a true positive).

- **Nonexistence of rule 2.** As illustrated in Figure 7, rule 2, which corresponds to the cases where the class variable is independent of the features, is extremely rare. This rule only starts to emerge for permutations superior to 0.2 in the diabetes dataset, and are nonexistent in the cancer dataset. This suggests that permutation variances of 0.1 do not point towards unrealistic and erroneous classifications. Figure 9 shows an example of an unreliable prediction that was found in the testset of the diabetes dataset, using a variance of $\epsilon = 0.25$ and the respective LIME and SHAP interpretations. Note that this specific datapoint corresponds to a misclassification (a false positive).
- **Contrast effects, rule 3, majorly coincide with misclassifications.** When a datapoint falls very close to the decision boundary, then when permuting that datapoint, there can be a significant statistical distribution of the predictions falling on the region

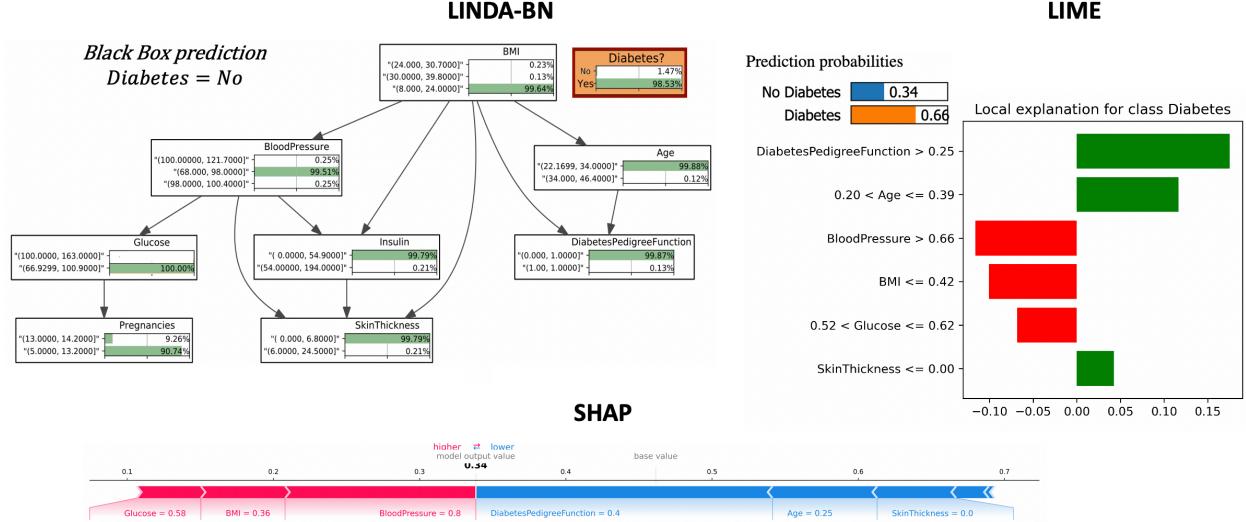


Figure 9: Misclassification in accordance with rule 2, an unreliable prediction (a false positive).

of the decision boundary of the opposite class. According to the analysis in Table 2, in the Diabetes dataset, rule 3 majorly occurred in the set of false positive points, that is datapoints that were misclassified. But there is also a significant percentage of datapoints in the set of true positives. Although these points were correctly classified, the contrast effect that is captured by the proposed interpretable model might indicate that these are cases correctly classified near the decision boundary. Thus, the decision-maker should be aware that although there was a correct classification, this classification might not have occurred due to the appropriate input feature values. Figure 10 shows an example of a rule 3 in the Diabetes dataset for $\epsilon = 0.1$. When comparing with LIME and SHAP, one can notice that it is not very clear that this datapoint represents a misclassification.

- **Uncertainty in predictions, rule 4, majorly coincide with misclassifications.**

In the experiments that were performed, using $\epsilon = 0.1$, Table 2 shows that the datapoints showing higher uncertainty in the likelihood of the class variable are mostly present in the sets of the false positives and the false negatives. In other words, in the set of misclassified datapoints. This is more noticeable in the diabetes dataset, in which the black-box predictor achieve an average accuracy of 73.8% where there are more misclassified datapoints. On the other hand, when one looks at the breast cancer dataset,

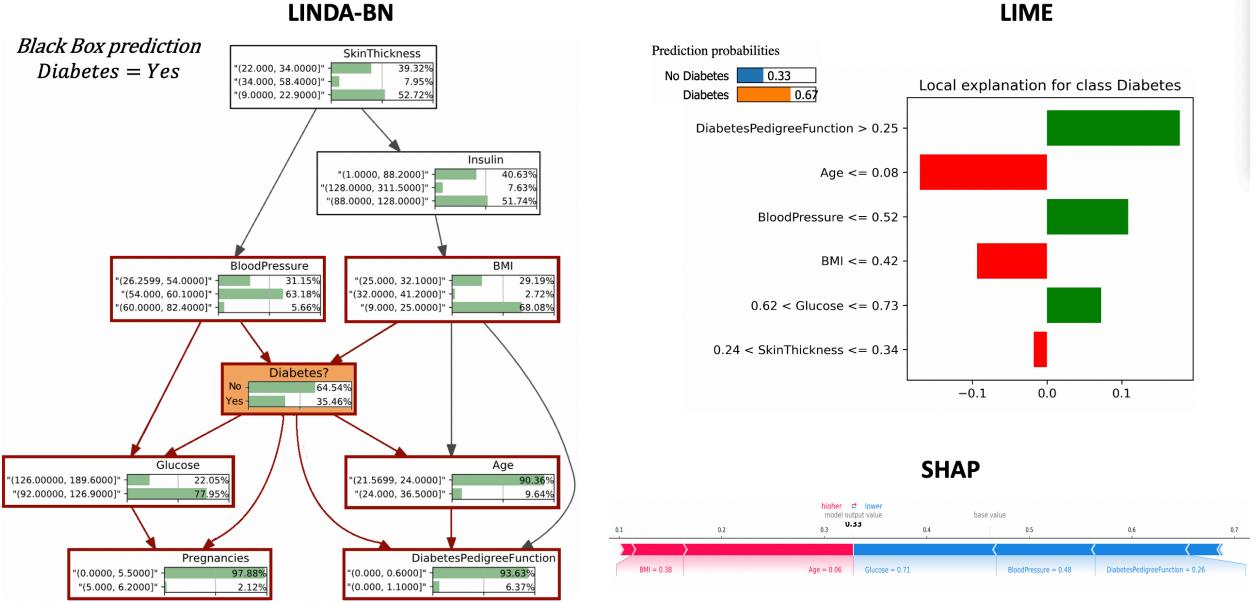


Figure 10: Misclassification in accordance with rule 3, a contrast effect.

since there were almost no misclassified datapoints (accuracy of 0.9840), the percentage of datapoints falling in rule 4 are nearly zero. Figure 11 illustrates an example of a false positive datapoint, in which the interpretable model show maximum uncertainty in the class variable node. In terms of LIME and SHAP, it is hard to identify any principles that could point the decision-maker about a possible misclassification.

In the next section, we describe how more complex decision problems are addressed by the proposed interpretable model.

4.4. Interpretations for Complex Decision Scenarios

For small decision problems (at most 10 random variables), the proposed LINDA model displays the full interpretable network. For more complex decision problems, however, this would become unreadable for a human decision-maker. The breast cancer dataset is an example of such complex decision problem that contains a set of 30 features, which are mapped into random variables. This results in a graphical structure too complex for any human to analyse. For such datasets, the proposed LINDA model provides the decision-maker a Markov Blanket of the variable of interest, instead of the full local interpretable Bayesian

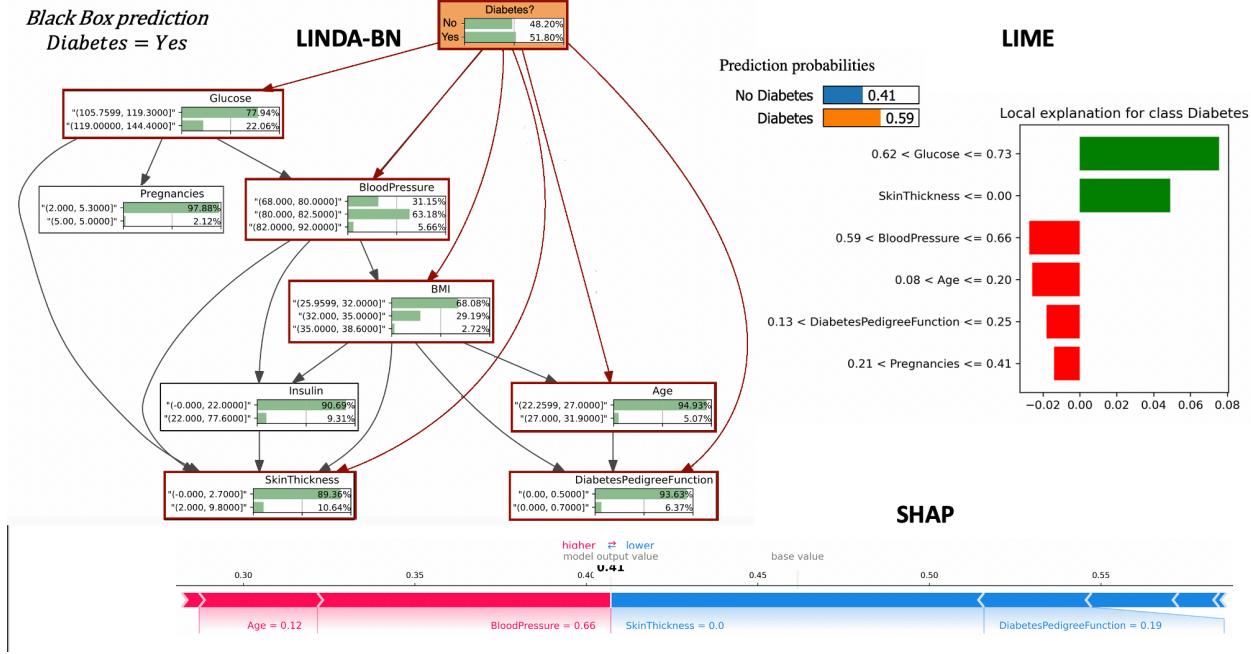


Figure 11: Correct classification in accordance with rule 4, uncertainty in predictions (true positive).

network, together with information about in which rule the network pattern corresponds to and respective marginal probabilities.

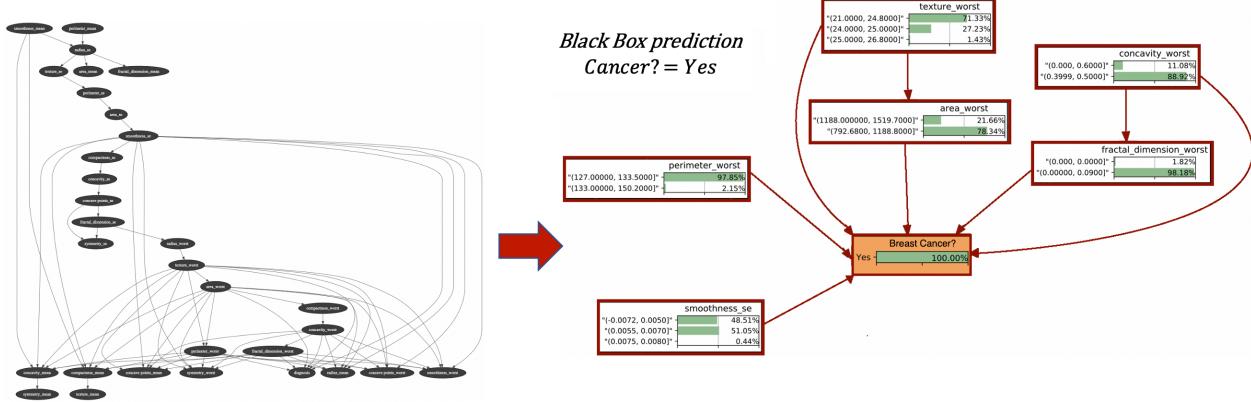


Figure 12: Markov blanket representation of a local interpretable Bayesian network with 30 nodes for the breast cancer dataset. The Markov blanket is in accordance with rule 1 and represents a correct classification.

This representation enables the summarisation of information, enabling a fast and compact data driven interpretation of a local datapoint. Figure 12 shows an example of an interpretable network that was extracted out of a true positive datapoint. The complexity of

the network does not enable any human interpretations to take place. However, when looking at the Markov blanket together with the marginal probabilities of the random variables, then one can clearly identify a common-effect structure in which six features directly influence the class variable and contribute to its value. Moreover, the statistical distribution of the permutations shows a full confidence in the diagnosis, suggesting that the datapoint falls within rule 1 and consequently there is a high confidence that it is a correct prediction. The depth of this Markov blanket can potentially be extended to different depths, depending on the decision-maker’s needs (for example. a normal person would be satisfied with the Markov blanket in Figure 12, however a medical doctor would probably explore other depths and analyse the relationship between more variables and their indirect influences towards the class variable).

5. Conclusions

In this paper, we proposed a new post hoc interpretable framework, the *Local Interpretation-Driven Abstract Bayesian Network* (LINDA-BN). This framework consists in learning a Bayesian network as an approximation of a black-box model from a statistical distribution of predictions from a local datapoint.

The major contribution of the proposed framework is the ability to identify four different rules which can inform the decision-maker about the confidence level in a given prediction of a specific datapoint. As such, the interpretations provided in our approach can help the decision-maker assess the reliability of predictions learned by a black-box model. These rules correspond to the different patterns that can be found in the learned Bayesian network, and they are summarised as follows:

- Rule 1 - High confidence in predictions: a common-effect structure in which the features are directly influencing the class and the maximum likelihood of the class is close to one, suggesting a correct classification.
- Rule 2 - Unreliable predictions: when the class variable is independent from the features, suggesting a misclassification

- Rule 3 - Contrast Effects: when the maximum likelihood of the class variable in the learned Bayesian network favours a class opposite to the black-box model, suggesting a misclassification.
- Rule 4 - Uncertainty in the predictions: when the likelihood of the class variable has very high levels of uncertainty, suggesting that the decision-maker should assess the network in order to understand if the features are supporting the class.

Experimental findings showed that rules 3 and 4 usually occurred in sets of false positives and false negatives, suggesting that the proposed framework might provide a possible approach to identify misclassifications in black-box models. On the other hand, the correct classifications (true positives and true negatives), were mostly associated with rule 1 with common-effect graph structures and maximum likelihood in the class variable close to one, again providing a potential method to identify correct classifications and promote trust in the decision-maker.

For future work, we would like to extend the proposed approach from an interpretable framework to an explainable one. This majorly consists in converting the symbolic rules proposed in this study into explainable arguments that could communicate the decision-maker *why* a certain prediction was computed out of a black-box and the reasons of *why / why not* a decision-maker should trust in the predictions.

References

- [1] W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Interpretable machine learning: definitions, methods, and applications, CoRR abs/1901.04592 (2019).
- [2] Q. V. Liao, D. M. Gruen, S. Miller, Questioning the AI: informing design practices for explainable AI user experiences, CoRR abs/2001.02478 (2020).
- [3] R. Guidotti, et al., A survey of methods for explaining black box models, ACM Computing Survey 51 (2018) 93:1–93:42.
- [4] H. Lakkaraju, et al., Faithful and customizable explanations of black box models, in: Proceedings of the 2019 AAAI Conference on AIES 2019, 2019, pp. 131–138.

- [5] Z. C. Lipton, The mythos of model interpretability, *CACM* 61 (2018) 36–43.
- [6] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning, arxiv: 1702.08608 (2017).
- [7] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Müller, Causability and explainability of artificial intelligence in medicine, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 9 (2019) e1312.
- [8] M. Siering, A. V. Deokar, C. Janze, Disentangling consumer recommendations: Explaining and predicting airline recommendations based on online reviews, *Decision Support Systems* 107 (2018) 52 – 63.
- [9] B. Kim, J. Park, J. Suh, Transparency and accountability in ai decision support: Explaining and visualizing convolutional neural networks for text information, *Decision Support Systems* 134 (2020) 113302.
- [10] M. T. Ribeiro, S. Singh, C. Guestrin, "Why Should I Trust You?": Explaining the predictions of any classifier, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.
- [11] S. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: *Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS)*, 2017.
- [12] M. A.-M. Radwa Elshawi, Youssef Sherif, S. Sakr, Interpretability in healthcare a comparative study of local machine learning interpretability techniques, in: *Proceedings of IEEE Symposium on Computer-Based Medical Systems (CBMS)*, 2019.
- [13] M. Stiffler, A. Hudler, E. Lee, D. Braines, D. Mott, D. Harborne, An analysis of the reliability of lime with deep learning models, in: *Proceedings of the Distributed Analytics and Information Science International Technology Alliance*, 2018.
- [14] H. F. Tan, K. Song, M. Udell, Y. Sun, Y. Zhang, Why should you trust my interpretation? understanding uncertainty in lime predictions, 2019.

- [15] M. T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, in: Proceedings of the 32nd AAAI International Conference on Artificial Intelligence, 2018.
- [16] L. S. Shapley, A value for n-person games, *Rand coporation* (1952) 15.
- [17] E. Strumbelj, I. Kononenko, Explaining prediction models and individual predictions with feature contributions, *Knowledge and Information Systems* 41 (2013) 647–665.
- [18] A. C. Miller Janny Ariza-Garzón, Javier Arroyo, M.-J. Segovia-Vargas, Explainability of a machine learning granting scoring model in peer-to-peer lending, in: Proceedings of IEEE Access, 2020.
- [19] A. B. Parsa, A. Movahedi, H. Taghipour, S. Derrible, A. (Kouros)Mohammadian, Toward safer highways, application of xgboost and shap for real-time accident detection and feature analysis, *Accident Analysis & Prevention* 136 (2020) 105405.
- [20] S. Wachter, B. Mittelstadt, C. Russell, Counterfactual explanations without opening the black box: Automated decisions and the gdpr, 2018.
- [21] C. T. Ramaravind K. Mothilal, Amit Sharma, Examples are not enough, learn to criticize! criticism for interpretability, in: Proceedings of the 2020 Conference on Fairness, Accountability, and TransparencyJanuary, 2020.
- [22] J. Pearl, The seven tools of causal inference, with reflections on machine learning, *Communications of ACM* 62 (2019) 7.
- [23] L. Bottou, J. Peters, J. Quiñonero-Candela, D. X. Charles, D. M. Chickering, E. Portugaly, D. Ray, P. Simard, E. Snelson, Counterfactual reasoning and learning systems: The example of computational advertising, *Journal of Machine Learning Research* 14 (2013) 3207–3260.
- [24] F. D. Johansson, U. Shalit, D. Sontag, Learning representations for counterfactual inference, 2016.
- [25] P. Schulam, S. Saria, Reliable decision support using counterfactual models, 2017.

- [26] E. C. Neto, Towards causality-aware predictions in static machine learning tasks: the linear structural causal model case, 2020.
- [27] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers, 1988.
- [28] S. Russel, P. Norvig, Artificial Intelligence: A Modern Approach, Pearson Education (3rd Edition), 2010.
- [29] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, The MIT Press, 2009.
- [30] M. Scutari, C. Vitolo, A. Tucker, Learning bayesian networks from big data with greedy search: computational complexity and efficient implementation, *Statistics and Computing* 29 (2019) 1095–1108.
- [31] D. Heckerman, D. Geiger, D. Maxwell, Learning bayesian networks: The combination of knowledge and statistical data, *Machine Learning* 20 (1995) 197–243.
- [32] D. Heckerman, A Tutorial on Learning with Bayesian Networks, Technical Report, Microsoft Research Advanced Technology Division, Microsoft Corporation, 1995.
- [33] D. M. Chickering, D. Heckerman, Learning Bayesian networks is NP-hard, Technical Report, Tech. Rep. MSR-TR-94-17, Microsoft Corporation, 1994.
- [34] D. M. Chickering, Learning Bayesian Networks is NP-Complete, Springer New York, 1996, pp. 121–130.
- [35] D. Gabbay, J. Woods, Advice on abductive logic, *Logic Journal of the IGPL* 14 (2006) 189–219.
- [36] S. Piri, D. Delen, T. Liu, A synthetic informative minority over-sampling (simo) algorithm leveraging support vector machine to enhance learning from imbalanced datasets, *Decision Support Systems* 106 (2018) 15–29.