Data & Knowledge Engineering 68 (2009) 886–904

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Explaining instance classifications with interactions of subsets of feature values

E. Štrumbelj *, I. Kononenko, M. Robnik Šikonja

University of Ljubljana, Faculty of Computer and Information Science, Tržaška cesta 25, 1000 Ljubljana, Slovenia

ARTICLE INFO

Article history:
Received 3 July 2008
Received in revised form 24 January 2009
Accepted 27 January 2009
Available online 5 February 2009

Keywords:
Data mining
Machine learning
Knowledge discovery
Visualization
Classification
Explanation

ABSTRACT

In this paper, we present a novel method for explaining the decisions of an arbitrary classifier, independent of the type of classifier. The method works at the instance level, decomposing the model's prediction for an instance into the contributions of the attributes' values. We use several artificial data sets and several different types of models to show that the generated explanations reflect the decision-making properties of the explained model and approach the concepts behind the data set as the prediction quality of the model increases. The usefulness of the method is justified by a successful application on a real-world breast cancer recurrence prediction problem.

1. Introduction

In recent years, there has been a growing use of data mining and machine learning outside the computer science community. Such methods are being integrated into decision support systems for fields such as finance, marketing, insurance, and medicine. Another partial motivation for our work is the need to support medical decisions, more precisely breast cancer recurrence prediction. We will use this example as a short introduction to data mining, modeling, and, finally, the explanation of models, which is the main topic of this paper.

A typical decision support scenario usually starts with a *domain* and a problem. In our medical example, the problem is whether or not the patient's cancer will recur. In general, the true concepts behind the domain are unknown, but oncologists have a certain degree of expert knowledge. They have also been gathering patients' data at the time of surgery and recording recurrence through patient follow up, all in an effort to gain some additional insight into the concepts that drive breast cancer recurrence. Each patient is therefore an *instance* from our domain and is described by several *features* (for example age, tumor size, etc.) and a *class* value or class label. In our case we have only two possible class values: recurrence, no recurrence. The set of all recorded instances is often referred to as the *data set*, which is where data mining and machine learning begin. The first phase is typically data preprocessing, where we address missing values, remove useless instances, possibly discretize continuous features, and address other similar issues. Once the data set has been preprocessed, it can be used to train, test, and choose the best performing prediction model. This can be combined with feature selection [8,17] to reduce the features to those relevant for the problem. Given that our problem is a classification problem, we can select from a variety of different classifiers, some of which will be more suitable and some less suitable for

* Corresponding author.
E-mail addresses: erik.strumbelj@fri.uni-lj.si (E. Štrumbelj), igor.kononenko@fri.uni-lj.si (I. Kononenko), marko.robnik@fri.uni-lj.si (M. Robnik Šikonja).

0169-023X/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.datak.2009.01.004

the problem. From these models, we pick the one that gives the best results and is therefore the most suitable model for our task. This model can be given to oncologists, who can use it as a black-box that can predict the probability of breast cancer recurrence for new patients. However, the task does not end here, because physicians are reluctant to use or even refuse to use these models, despite the fact that models outperform physicians in several medical prediction tasks [13]. Their reluctance to make a decision based only on a single probabilistic output is understandable, given their huge responsibility and the importance of the decision. Similarly, users from other fields may also be reluctant or may require additional insight into the model's decision. That is why there is a growing need to enhance the predictions with additional information – an *explanation*.

1.1. Existing explanation methods

Some models already have a transparent decision-making process from which it is easy to extract an explanation. For example, a decision tree's decision for an instance can be explained by observing the rules that lead from the root to the leaf that contains the instance. Bayesian networks are also an example of a highly interpretable model and their interpretability can be further improved [7]. For most other, less transparent models, model-dependent approaches have been developed. For example, Breiman provided additional tools for explaining the decisions of his Random forests algorithm [4], and a lot of work has been done on explaining the decisions of artificial neural networks [24], which are one of the least transparent models. The ExplainD [22] framework provides explanations for additive classifiers. Nomograms [18] are also used for visualization and explanation of decisions and were applied to naive Bayes [19] and, in a limited way, to *SVM* (Support Vector Machines) [11]. The interpretability of some models can also be improved by systematically reducing the number of training instances (see, for example [6]). Considering that for almost any given model, we already have a model-dependent explanation method, why do we even need a model-independent explanation method? The main reason is *consistency of explanation*. The problem of model-dependent methods is that once the model is replaced with a model of a different type, the explanation method also has to be replaced. This is very undesirable from the user's perspective, because it often takes a lot of additional time and effort to get used to a new explanation method. Also, it is quite common to have several different models performing different tasks in the same decision support system. With a model-independent explanation method, we can explain their decisions to the user in a uniform way.

As far as we know, there are two existing model-independent methods for explaining a model's decision for an instance. One is proposed by Robnik-Šikonja and Kononenko [20] (we will refer to this method as *EXPLAIN*) and the other by Lemaire et al. [16]. While there are several differences between these two methods, they both have the same approach. Both methods compute the influence of a feature value by changing that value and observing the changes in the model's output. Bigger changes in the output imply that the feature value was more important for the prediction, and vice versa. However, because both methods observe only a single feature at a time, they have a huge flaw, which can be illustrated by simple disjunction. Let A_1 and A_2 be binary features from the domain with a binary class and the following concept: class $= A_1 \vee A_2$. Now let us explain the instance $A_1 = 1, A_2 = 1$. Obviously the decision is $1 \vee 1$ equals 1. By changing A_1 to 0 we get $0 \vee 1$, which still equals 1. Because the prediction did not change, both methods would conclude that $A_1 = 1$ is irrelevant. Symmetrically, A_2 will be marked as irrelevant as well. This flaw is not reserved just for disjunction but is present whenever there is redundancy among the features or their values. As a result, the explanations generated by existing methods are often inaccurate and misleading, which, in turn, makes them more difficult to trust.

1.2. A new instance explanation method

In this paper, we propose a new model-independent explanation method *IME* (Interactions-based Method for Explanation) that is designed to deal with the flaws of existing model-independent methods. As *IME* is a method for explaining classifier decisions for an instance, we assume that we have an arbitrary classifier trained on a data set, a new instance from the given domain, and a class value. The classifier makes a prediction, in other words an assessment of the class value's probability, knowing the feature values of the instance. There is usually a difference between this assessment and the prior probability of the class value (i.e., if no feature values are known). The question is: *Which features influenced the classifier's decision and caused this difference in prediction? IME* provides the answer by assigning to each feature value a real number – a *contribution*. The sum of these contributions equals the difference in prediction, and the size of each contribution is proportional to the feature value's influence on the decision. A positive contribution indicates that the value contributes to the class value, and a negative contribution indicates that the value speaks against the class value.

It is obvious from the disjunction example that we may overlook an important part of the model's decision-making process, unless we look at every possible combination of feature values. Therefore, to avoid missing something, we observe how the prediction changes for each subset of the *power set* of feature values. These changes are then combined to form a contribution for each feature value. Dealing with the whole power set results in an exponential time complexity, but, there are several reasons why the method is already a valuable contribution. *IME* is able to deal with various concepts (even where existing model-independent methods would fail) and generate useful and intuitive explanations for a variety of different models, as we will illustrate on several artificial data sets. The method is especially useful when the user's trust is a priority and a precise explanation is needed. Because the generated contributions are implicitly normalized, it makes it easy to com-

pare them across instances and even across different models. A real-life breast cancer recurrence prediction problem is used as an example to illustrate the method being successfully applied in practice. Finally, we discuss how the exponential time complexity of *IME* can be overcome.

The paper is divided into six sections. In Section 2, we formally define the *IME* method. In Section 3, we introduce several artificial data sets and use them to empirically test the method. Section 4 is dedicated to the analysis of the intuitiveness and usability of the method's explanations. In Section 5, we discuss the method's high time complexity and point out possible solutions. With Section 6, we conclude the paper and offer some ideas for further work.

2. The definition of the method

Let $A_1, A_2, \ldots, A_n$ be a set of features, $\Re_i$ the ith feature's value range, and $c \in C$ the class value we are explaining from the finite set of class values for this domain, C. Let $\mathscr{X} = \Re_1 \times \Re_2 \times \cdots \times \Re_n$ be the instance space. Let $\mathscr{X}_Q = \Re_{j_1} \times \Re_{j_2} \times \cdots \times \Re_{j_m}, j_i \in Q$ be a subspace of the instance space where only the features from $Q \subseteq \{1, 2, \ldots, n\}$ are known. Note that j_i are disjoint members of Q. Let $\widehat{\mathscr{X}} = \bigcup_{W \subseteq \{1,2,\ldots,n\}} \mathscr{X}_W$ be the union of all possible subspaces. A probabilistic classifier $h : \widehat{\mathscr{X}} \to [0,1]^m$ maps an instance $x_Q \in \mathscr{X}_Q$ to a set of m probabilities, one for each class value ($m = |C|$). Given that we are interested in explaining only a single class value at a time, we can write $h : \widehat{\mathscr{X}} \to [0,1]$, where $h(x_Q)$ is the classifier's approximation of the probability $P(C = c|x_Q)$. Note that we at this point assume that each classifier is able to predict the class value or class probability for x_Q, an instance where arbitrary features are omitted. However, in practice this is not always true, so ways of approximating these *marginal predictions* will be discussed later on.

Given x, an instance we are trying to explain, there is a difference between a prediction using all features and a prediction using no features. This difference is caused by the influence of the feature values on instance x, and it is this *prediction difference* that we are trying to explain:

$$\Delta_{\{1,2,\ldots,n\}} = h(x_{\{1,2,\ldots,n\}}) - h(x_\varnothing) \tag{1}$$

Note that Δ-terms are instance-dependent but we omit it in the notation to simplify the presentation. We can generalize (1) to an arbitrary instance subspace:

$$\Delta_Q = h(x_Q) - h(x_\varnothing) \tag{2}$$

The difference Δ_Q is not only a result of the influence of individual feature values but also a result of how feature values interact. In fact, observing a set Q of feature values together might contribute something to the decision that could not be seen when observing any true subset of Q. We refer to such contributions as *interaction contributions*. Let Δ_Q be the sum of all such interaction contributions:

$$\Delta_Q = \sum_{W \subseteq Q} \mathscr{I}_W \tag{3}$$

With (3), we decomposed the difference in prediction into $2^{|Q|}$ parts, one for each possible subset of feature values. Now, we derive our definition of an interaction contribution $\mathscr{I}_Q$ of feature values from Q. We additionally define that the contribution of an empty-set interaction is zero, which results in the following recursive definition:

$$\begin{aligned} \mathscr{I}_Q &= \Delta_Q - \sum_{W \subset Q} \mathscr{I}_W \\ \mathscr{I}_{\{\}} &= 0 \end{aligned} \tag{4}$$

The 2^n interaction contributions that we get for our instance x already provide an explanation. However, this explanation can be difficult to understand due to the large number of possible subsets. Therefore, we first assign a single contribution to each feature value. This is done by dividing each interaction contribution into as many parts as there are feature values involved in that interaction. Each part is then assigned to the corresponding feature value's contribution. Let π_i bet the contribution of the ith feature's value:

$$\pi_i = \sum_{W \subseteq \{1,2,\ldots,n\} \wedge i \in W} \frac{\mathscr{I}_W}{|W|} \tag{5}$$

One might argue that less probable values play a more important role in an interaction and thus deserve a larger share of the interaction, so we may obtain more informative results by dividing the interaction based on the feature value's prior probabilities. However, by using additional information such as feature value probabilities, we bypass the model that we are explaining and compromise the explanation by forcing our external view onto it. Without using any prior knowledge, dividing into parts of equal size is the only symmetrical way of completely assigning an interaction contribution to the contributions of involved feature values. The other argument against this division is that it assumes that feature values are either independent or equiprobable. However, as we will show in three illustrative examples, the interpretation of dependencies is left to the model that we are explaining, which is arguably what an explanation method should do. In other words, if the model has successfully learned a dependency among feature values, then this will also be reflected in the explanation.

2.1. Illustrative examples

Our first illustrative example will be a domain with the disjunction concept. Let A_1, A_2, A_3 be three binary features, $C = \{0,1\}$ a binary class, and $\mathcal{X} = \{0,1\} \times \{0,1\} \times \{0,1\}$ our instance space. The concept is $C = A_1 \vee A_2 \vee A_3$. Let $h : \widehat{\mathcal{X}} \rightarrow [0,1]$ be an (ideal) Bayesian classifier which was trained on a data set where all feature value combinations are equiprobable, as seen in Table 1. Our goal is to explain how h classifies the instance $a' = \{1,1,0\}$ from the perspective of class value 1. In other words, how did the three feature values contribute to the model's prediction $P(C = 1|a') = 1$?

The prior probability of class 1, or $h(a'_\emptyset)$, is $\frac{7}{8}$ (i.e., out of eight instances in this data set, seven have class value 1). As long as we take into account at least one of the first two feature values, the predicted probability will be 1. In other words, $h(a'_W) = 1$ whenever at least one of the first two features is included in W. Therefore, according to (2), the following Δ-terms have the same value: $\Delta_{\{1\}} = \Delta_{\{2\}} = \Delta_{\{1,2\}} = \Delta_{\{1,3\}} = \Delta_{\{2,3\}} = \Delta_{\{1,2,3\}} = 1 - \frac{7}{8} = \frac{1}{8}$. The term $\Delta_{\{3\}}$ is different, because the predicted probability is now equal to the probability of at least one of the remaining values being one. In other words, if we observe only the third column and only those rows where the value of the third feature is 0, we narrow the data set down to four instances, three of which have class value 1. Therefore, $h(a'_{\{3\}})$ equals $\frac{3}{4}$ and $\Delta_{\{3\}} = \frac{3}{4} - \frac{7}{8} = -\frac{1}{8}$.

We can now use (4) to calculate the interaction contributions. Single-feature interaction contributions are equal to the corresponding Δ-terms:

$$\mathcal{I}_{\{1,2\}} = \Delta_{\{1,2\}} - \mathcal{I}_{\{1\}} - \mathcal{I}_{\{2\}} = \frac{1}{8} - \frac{1}{8} - \frac{1}{8} = -\frac{1}{8}$$

$$\mathcal{I}_{\{1,3\}} = \mathcal{I}_{\{2,3\}} = \frac{1}{8} - \frac{1}{8} + \frac{1}{8} = \frac{1}{8}$$

$$\mathcal{I}_{\{1,2,3\}} = \Delta_{\{1,2,3\}} - \mathcal{I}_{\{1,2\}} - \mathcal{I}_{\{1,3\}} - \mathcal{I}_{\{2,3\}} - \mathcal{I}_{\{1\}} - \mathcal{I}_{\{2\}} - \mathcal{I}_{\{3\}} = -\frac{1}{8}$$

Finally, the contributions of feature values are:

$$\pi_1 = \frac{\mathcal{I}_{\{1,2,3\}}}{3} + \frac{\mathcal{I}_{\{1,2\}} + \mathcal{I}_{\{1,3\}}}{2} + \mathcal{I}_{\{1\}} = \frac{1}{12}$$

$$\pi_2 = \frac{1}{12}$$

$$\pi_3 = -\frac{1}{24}$$

The sum of all contributions equals the total difference between the predicted probability and the prior class probability ($\pi_1 + \pi_2 + \pi_3 = \frac{3}{24} = \frac{1}{8} = 1 -$ prior probability). Therefore, for instance a' both 1's speak favorably towards the class being 1 and $A_3 = 0$ speaks slightly less in favor of the opposite class, which is consistent with our intuitive explanation for this instance. As we can see, IME does not have difficulties dealing with disjunctive concepts.

Using the same domain, but explaining the instance $a'' = \{0,0,0\}$, we get the following contributions: $\{-\frac{7}{24}, -\frac{7}{24}, -\frac{7}{24}\}$. Up to now we have assumed that the feature values are both independent and equiprobable, but how would the contributions change if they were not? Let the probabilities of $A_1 = 1$, $A_2 = 1$, and $A_3 = 1$ be 0.9, 0.5, and 0.2, respectively. Now the contributions change to $\{-0.615, -0.255, -0.09\}$, which makes sense, because the feature value which was most likely to be 1 is assigned the largest negative contribution for being 0. The larger the probability of the value being 0, the smaller the negative contribution when the value is actually 0.

Finally, what if feature values are not even independent? Let A_1, A_2, A_3 again be three binary features and $C = \{0,1\}$ a binary class. The concept is $C = A_1 \wedge A_2 \wedge A_3$ and we train the classifier on the data set in Table 2, which contains a strong dependency between two features ($A_1 = 1 \Rightarrow A_2 = 1$). For example, on $a''' = \{1,1,1\}$ the explanation method generates the contributions $\{\frac{22}{72}, \frac{4}{72}, \frac{28}{72}\}$. The feature value $A_2 = 1$ is almost completely dependent on $A_1 = 1$, and it is assigned a very small contribution to reflect that. The remaining two feature values contribute roughly the same amount, although $A_3 = 1$ has a slightly larger contribution due to the fact that the other two feature values bring nothing new when observed together.

Table 1
A simple data set used for illustrative purposes. All combinations of features are equally probable.

A_1	A_2	A_3	C
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Table 2
A simple data set used for illustrative purposes. All combinations of features are not equally probable and there is a strong dependency between the first two features.

A_1	A_2	A_3	C
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	1	0	0
1	1	1	1
1	1	0	0
1	1	1	1

2.2. Approximating marginal predictions

Not all models have the ability to make marginal predictions (i.e., predict instances with omitted feature values), so we have to approximate these predictions. The approach used in this paper is similar to the wrapper approach [12] used in feature-selection. We approximate a model's marginal prediction $h(x_Q)$ by removing the features that are not in Q from the data set and retraining the model. This results in a new model h_Q and a smaller instance space $\mathscr{X}_Q$, where $x_Q \in \mathscr{X}_Q$. For model h_Q, instance x_Q has no missing features, so we can use it to approximate the marginal prediction: $h(x_Q) \approx h_Q(x_Q)$. The advantage of this approach is that it works for both continuous and discrete features. The downside is that the model has to be retrained for each of the 2^n subsets of features.

Two other approaches for marginal prediction approximation are worthy of mention, but are not used in our method. The first approach involves assigning special *unknown values* or *NA*'s to feature values that we want to omit from the instance. This approach is limited to models that support the use of unknown values, which is the reason we can not use this approach in our generalized explanation method. The second approach approximates the omission of a feature value with an average expected prediction across all perturbations of the feature value, where each perturbation is weighted by its probability. Both Robnik-Šikonja and Kononenko [20] and Lemaire et al. [16] use variations of this approach in their model-independent explanation methods. Although it was used only for approximating the omission of a single feature, it can be generalized to an arbitrary number of features, with an exponentially growing number of perturbations. As the number of omitted feature values increases, so does the approximation error. The advantage of this approach is that the model does not have to be retrained. However, continuous features have to be discretized because we can only try a finite number of feature value combinations. Note that this discretization is performed only for the purpose of explanation and does not affect the classification.

3. Does the explanation reflect the model?

While the task of a model is to describe the domain, the explanation method's task is to describe the model's interpretation of the domain. Ideally, the better the model is at recognizing the domain concepts, the higher the quality of the explanation should be, and vice versa. In other words, the explanation of a poor-performing model may not reflect the concepts behind the domain, and the explanations of a well-performing model should reflect at least some concepts behind the domain.

The quality of the model's interpretation can be described by its *prediction quality* and the closeness of the explanation to the actual concepts behind the domain, known as *explanation quality*. This calls for several definitions. To measure a model's *prediction quality*, we use the Brier score [5]. The Brier score, b, is the squared deviation between the predicted probability for the class value, p, and the actual outcome p_a: $b = (p - p_a)^2$. The lower the Brier score, the better the prediction; therefore, it measures the classifier's ability to make precise probabilistic assessments. Our method uses the classifiers' predicted probabilities, which makes the Brier score more appropriate than other well-known prediction quality measures such as prediction accuracy (measures the classifier's ability to correctly classify an instance) or Area Under the ROC Curve (measures the classifier's ability to distinguish between instances with different class values).

We define *explanation quality* by first defining the *optimal explanation* of an instance. The optimal explanation of an instance is a set of n contributions $\pi_1, \pi_2, \ldots, \pi_n$ that we obtain by generating an explanation using an optimal Bayesian classifier. For example, all the illustrative explanations generated in Section 2.1 were generated using a Bayesian classifier and are therefore optimal explanations for those instances and those data sets. This is based on the assumption that an explanation method best describes the concepts behind the domain when explaining a model that optimally learns the concepts. Note that the optimal explanation is method-dependent, so different explanation methods would produce different optimal explanations for the same instance. We measure the *explanation quality* of an explanation by measuring how much it differs from the optimal explanation. For this purpose we use a Euclidean distance, d, between the generated contributions and the contributions from the optimal explanation.

We will use several artificial data sets and different types of classifiers to empirically show that *IME explanation quality* correlates with the model's *prediction quality*. Note that the use of artificial data sets, where all concepts are known, is absolutely necessary if we want to compute exact optimal explanations. Artificial data sets also allow us to test the method's ability to handle extreme examples of concepts that are uncommon in real-world data sets.

3.1. Models and artificial data sets

We tested the method using the following models: naive Bayes (*NB*), decision tree (*DT*), k-nearest neighbors (*kNN*), support vector machine with a polynomial kernel (*SVM*) and a multilayered feed-forward artificial neural network (*ANN*). See, for example, [14] for detailed descriptions of these learning algorithms.

The first five artificial data sets described in this section were introduced by Robnik-Šikonja and Kononenko [20] to test their explanation method. These data sets are designed so that each best suits a particular classifier. Therefore we know what to expect quality-wise, and we test to see if the explanation quality corresponds to the model's quality. Three additional data sets (disjunct, sphere, and random) are introduced to additionally test the method's ability to handle discrete and continuous disjunctive concepts and completely random features, respectively. Each data set consists of 2000 examples, half of which are used for training the model and half for testing it. All features are either binary or continuous with values from the $[0,1]$ interval, and, unless otherwise noted, 0 and 1 are the only two possible class values. On all data sets we assume we are explaining class value 1. We now briefly describe the artificial data sets:

condInd: This data set has eight binary features. The class value is also binary and both classes are equally probable. The four important features are equal to the class in 0.9, 0.8, 0.7 and 0.6 percent of the cases, respectively. The remaining four features are unrelated to the class. Each of the 16 possible instances from this data set has a different optimal explanation, which we omit due to space limitations. Because the features are conditionally independent given the class, *NB* is considered to be the most suitable for this data set.

xor: This data set has six binary features, three of which are unrelated to the class, and the class value indicates the parity of the remaining three important features. Noise was added to the class by reverting it in 10% of the cases so for each instance, the Bayesian classifier would assign either a 90% probability to class 1 (if there was parity) or a 10% to the class 1 (if there was no parity). Therefore, the difference between the true prior class probability, $\frac{1}{2}$, and the optimal predicted class probability is either $\frac{2}{5}$ or $-\frac{2}{5}$. There are three important features of equal importance, so the optimal explanation of an instance assigns $\frac{2}{15}$ to values of important features when parity exists and $-\frac{2}{15}$ when it does not. *DT* is considered most suitable for this data set as it can split the data on each feature and each leaf can then correspond to a different rule.

group: This data set has four continuous features. The three class values are scattered around group centers that are placed so that both important feature values have to be known to gain any knowledge about the class value. The remaining two features are unrelated to the class. The prior probability of class 1 is $\frac{1}{3}$, so the difference between the prior and the true predicted probability is either $\frac{2}{3}$ or $-\frac{1}{3}$. The optimal explanation of an instance assigns $\frac{1}{3}$ to each of the two important features when the class is 1 and $-\frac{1}{6}$ when it is either 0 or 2. The *kNN* model is considered most suitable for this data set because instances with the same class value are clustered together.

cross: This data set has four continuous features, two of which are unrelated to the class. The class value is 1 when $(I_1 - 0.5)(I_2 - 0.5) > 0$, where I_1 and I_2 are continuous features with values from the $[0,1]$ interval. The difference between the true prior class probability and the true predicted class probability is either $\frac{1}{2}$ (when the class is 1) or $-\frac{1}{2}$ (when the class is 0). There are two important features, so the true explanation of an instance assigns $\frac{1}{4}$ to each of them when the class is 1 and $-\frac{1}{4}$ when it is 0. *SVM* is considered best when dealing with such a data set because instances with different class values can be linearly separated in the *SVM*'s transformed feature space.

chess: This data set has four continuous features and represents a 4×4 chessboard where instances have either class value 1 if they fall on a black square or class value 0 if they fall on a white square. The two important features represent the x and y coordinates, and the remaining two features are unrelated to the class. Similar to the cross data set, the optimal explanation of an instance assigns $\frac{1}{4}$ to each of the two important features when the class is 1 and $-\frac{1}{4}$ when the class is 0. The *ANN* is considered most suitable for this data set, because the concept behind the data set is more complex and the remaining classifiers will have more difficulty due to the assumptions they make. Only *kNN* should have some success in predicting the class values of instances close to the center of each field, but not close to the edge.

sphere: This data set has five continuous features. It represents a sphere centered on point $C(0.5, 0.5, 0.5)$ and with a radius of 0.5. The three important features serve as three-dimensional coordinates, and the class value is 1 if and only if the instance lies within the sphere. The optimal explanation of an instance from this data set is more complex since it is a continuous function of the three important feature values. The user can refer to the Appendix for the derivation of individual Δ-terms for this data set.

disjunct: This data set has five features, two of which are unrelated to the class, and the class value is 1 if the value of any of the three important features is 1. The true explanation of an instance is a bit more complicated for this data set and depends on the number of ones and zeros. If all three important features are 1, then each contributes $\frac{1}{24}$. If one of them is 0, then $-\frac{1}{24}$ is assigned to that feature, and $\frac{1}{12}$ is assigned to the remaining two. If two features are 0, then they each contribute $-\frac{1}{12}$ and the remaining feature contributes $\frac{7}{24}$. If all the important features are 0, then their values contribute $-\frac{7}{24}$ each.

random: This data set has four continuous features, all of which are unrelated to the class. The optimal explanation of an instance would therefore assign 0 to every feature value.

3.2. Results

The complete test results are shown in Table 3, and the models that were most suitable for each of the first five data sets do in fact produce the best predictions (underlined). The best explanations are also generated on these models (underlined),

Table 3
Mean prediction quality ($\bar{b}$) and mean explanation quality ($\bar{d}$) across all test instances for each classifier and data set pair.

		Results				
		NB	*DT*	*kNN*	*SVM*	*ANN*
condInd	$\bar{b}$	0.068	0.079	0.088	0.077	0.099
	$\bar{d}$	0.029	0.082	0.123	0.088	0.083
xor	$\bar{b}$	0.252	0.090	0.097	0.256	0.098
	$\bar{d}$	0.233	0.033	0.149	0.231	0.056
group	$\bar{b}$	0.220	0.217	0.008	0.204	0.014
	$\bar{d}$	0.313	0.310	0.124	0.265	0.100
cross	$\bar{b}$	0.248	0.251	0.257	0.011	0.069
	$\bar{d}$	0.353	0.354	0.341	0.033	0.159
chess	$\bar{b}$	0.250	0.250	0.197	0.257	0.183
	$\bar{d}$	0.354	0.354	0.289	0.359	0.284
sphere	$\bar{b}$	0.142	0.085	0.122	0.013	0.091
	$\bar{d}$	0.281	0.188	0.220	0.075	0.185
disjunct	$\bar{b}$	0.014	0.000	0.000	0.000	0.000
	$\bar{d}$	0.038	0.130	0.085	0.096	0.058
random	$\bar{b}$	0.251	0.251	0.268	0.250	0.354
	$\bar{d}$	0.030	0.000	0.164	0.025	0.206

with the exception of the *groups* domain, where *ANN* produces a slightly better explanation despite being slightly out-preformed prediction-wise by the *kNN* model.

In Fig. 1, we plot the Brier score and $\bar{d}$ pairs from Table 3 for four data sets. Note that $\bar{d}$ is the mean *explanation quality* across all test instances. As we can see, there is no case of a better explanation being generated on a worse-performing model

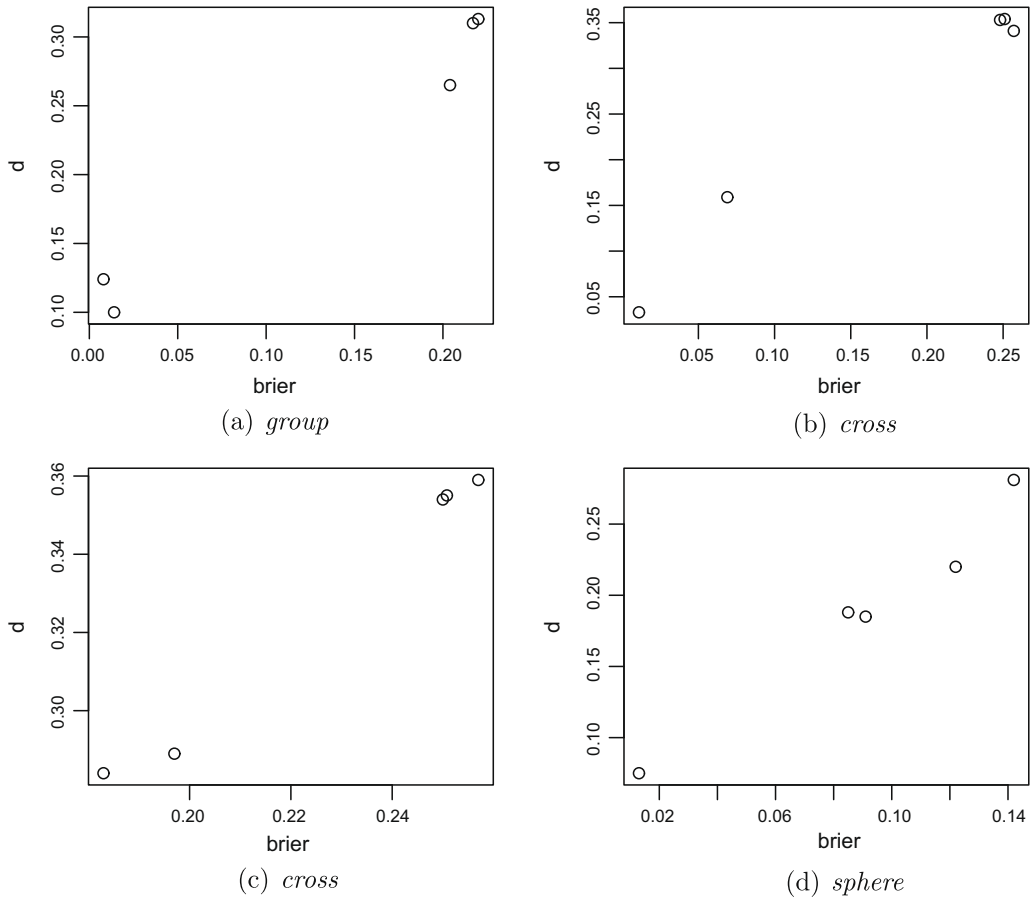


Fig. 1. The dependency between the quality of prediction and the quality of explanation on four different data sets and all five models. A low-quality model never produces a high-quality explanation or vice versa. These results suggest that explanation quality increases with increasing prediction quality, regardless of data set of type of model used.

on any data set. There is also no case of a good explanation being generated for a poor-performing model. In fact, there is an almost linear connection between prediction and explanation quality. This implies that with increasing prediction quality the explanations near the optimal explanations. In practice this means that the explanation of a high-quality model would not only reflect how the model works, but also the concepts behind domain itself. Note that an anomaly appears in the *disjunct* data set results, where models with a perfect Brier score generate sub-optimal explanations. This is possible, because we do not evaluate the Brier score of the sub-models that we build to obtain marginal predictions, yet we use their outputs. To remedy this, we could combine the Brier scores of all the sub-models built and get a more precise indicator of the model's quality.

4. Do the explanations make sense?

In the previous section, we have shown that the explanations reflect the model. However, the explanations would be of no use if they were not also intuitive and informative. We will use several examples to emphasize the usefulness of the explanations generated by our method. In parallel, we will also introduce different types of visualization. Fig. 2a is a simple visualization of the generated contributions for an instance from the first of the famous Monk data sets [23]. At the top, we have the name of the data set, the name of the model (the decision of an artificial neural network is being explained), the predicted class probability for this instance, and the actual class value. On the left-hand side of the visualization, we have the names of the features, and on the right-hand side, we have the feature values for this instance. The bars represent the contributions of individual feature values. The concept behind the Monk1 data set is that the class value equals one if and only if $A_1 = A_2$ or $A_5 = 1$. In our instance, both conditions are true, and the classifier correctly predicted class 1 with the probability of 1. The explanation correctly reflects that features A_1 and A_2 are of nearly equal importance and that $A_5 = 1$ is the most important contributor to the classifier's decision. The remaining feature values are assigned insignificant contributions, which reflects the fact that they are irrelevant for the decision. Note that the contributions of A_1 and A_2 are not entirely equal due to the fact that their values are not represented in the training data set with the same frequency and always in the same context. If two features would be completely interchangeable (as is the case with the illustrative data set in Table 1 then the generated contributions would be equal as well.

In Fig. 2b we have another explanation for the same instance, but, this time, we are explaining the classification of the naive Bayes classifier. Similar to the artificial neural network, the naive Bayes correctly classifies the instance. However, the explanation reveals that only $A_5 = 1$ had a significant contribution to the decision. While this might at first sight seem incorrect, it actually correctly reflects how the naive Bayes classifier works. The naive Bayes classifier is based on the assumption that the features are conditionally independent given the class value; this assumption leaves it unable to learn the concept of equality. These two visualizations from Fig. 2 illustrate how the explanations can provide insight into specific properties of models and enable comparison among different types of models.

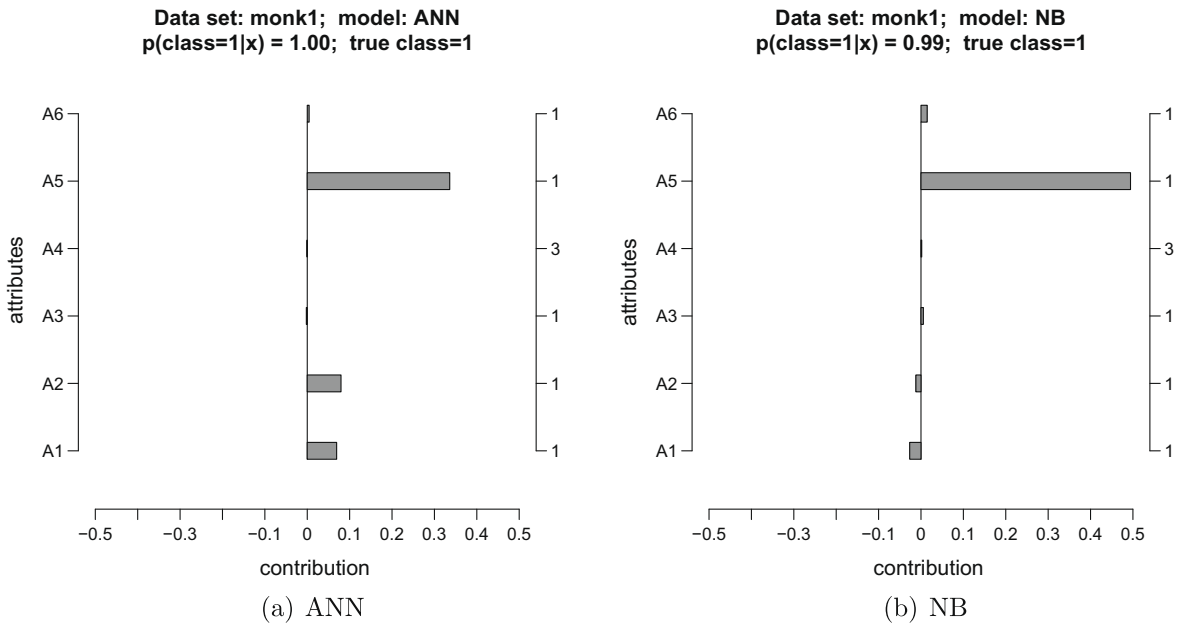
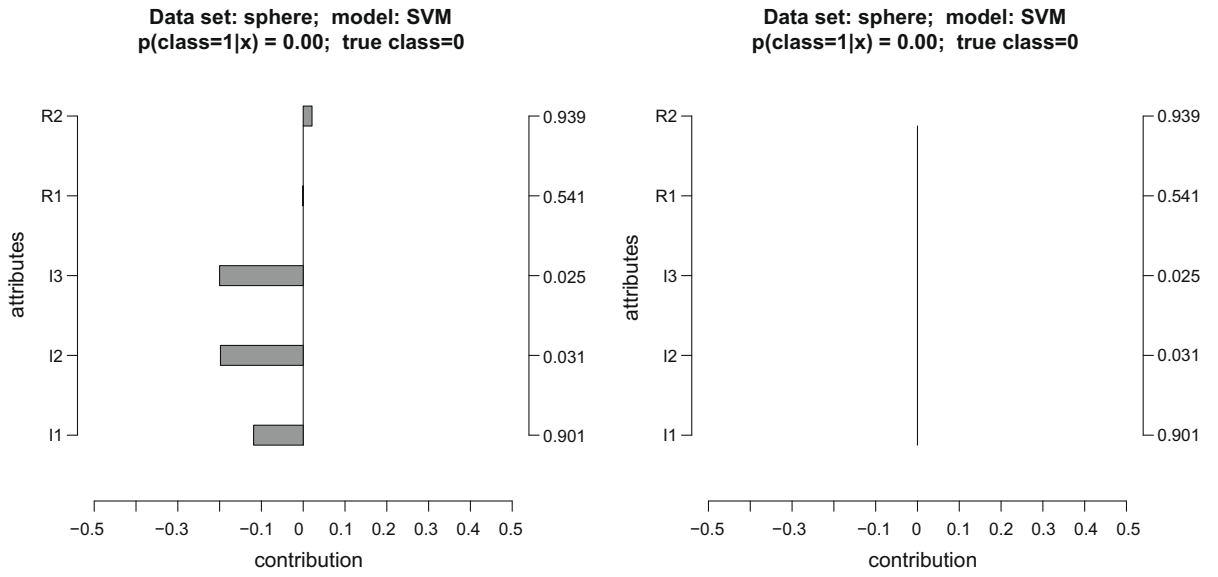


Fig. 2. Two explanation of the same instance from the Monk1 data set. On the left-hand side is the explanation of an artificial neural network's decision. On the right-hand side the naive Bayes model's decision is explained.



(a) IME explanation (b) EXPLAIN explanation

Fig. 3. Explanation of an instance from the sphere data set. The model being explained was a *SVM* classifier. On the left-hand side is the explanation generated by *IME* and on the right-hand side the explanation generated by the *EXPLAIN* method proposed by Robnik-Šikonja and Kononenko.

Fig. 3a is an explanation of an instance from the *sphere* data set, using the *SVM* classifier. The contributions can be either positive (reaching right) or negative (reaching left), depending on whether the feature value speaks for or against the class value. For example, the value 0.031 of the important feature I_2 makes it more likely that the instance lies outside the sphere and has a negative contribution. The value 0.939 of the random feature R_2 has an insignificant contribution because the feature is in fact unrelated to the class. Because the values of the three important features are far away from the center of the sphere, any two of the values would provide sufficient knowledge to classify the instance outside of the sphere (class value 0). This is an example of disjunction with continuous features. In Fig. 3b, we see the explanation generated using the *EXPLAIN* method, which incorrectly assigns a zero contribution to each feature value.

Finally, we may combine the contributions of individual feature values across several instances to get an overview of how the model interprets the data set. Fig. 4 is such a model explanation and is a result of combining contributions across all the test instances of the *sphere* data set for the *SVM*. The light bars represent the mean positive and mean negative contributions of individual feature values across all test instances. The darker bars represent the mean positive and mean negative contributions of each feature. The three important features are easily identified, as is the fact that values closer to the mean values (i.e., the center of the sphere) contribute more to the class value 1 and values farther away speak against class value 1. The *SVM* classifier's performance on this data set is excellent, so the visualization reflects the "spherical" nature of the data set.

4.1. Enhanced explanations

Up to this point, our visualizations contained only feature value contributions, which tell us how much each feature value contributes to the decision of the model. However, the user is often interested not only in the magnitude and direction of the feature value's contribution but also in possible dependencies and interactions between feature values. In such cases, the user can use an enhanced instance explanation.

In Fig. 5a, we have an enhanced instance explanation for the Monk1 instance that we discussed in the previous section. Each feature value now has not only a bar indicating its contribution but also a vertical line. This vertical line is the value's contribution on its own (or $\mathscr{I}_{\{A_i\}}$) without any interactions with other feature values. Immediately, we notice that $A_5 = 1$ by itself would result in approximately the same certainty of class value 1 and that it has a smaller contribution when observed with other feature values. On the other hand, $A_1 = 1$ and $A_2 = 1$ have only a small negative contribution when each is observed on its own. Therefore, the vertical lines inform the user that there must be some interaction between these three feature values. The user can now use the list of interaction contributions that involve these three features to gain further insight into the model's decision (see Fig. 5a). The interactions are listed according to their absolute value, so we can immediately notice that the interaction of $A_1 = 1$ and $A_2 = 1$ has a large contribution for class value 1. On the other hand, the interaction of all three important feature values is equally negative, so all three together bring much less than the sum of their individual contributions would suggest. Now, we can easily come to the conclusion that $A_5 = 1$ tilts the decision to class value 1 (with probability of 1) by itself, that $A_1 = 1$ and $A_2 = 1$ together have the same effect, but that all three together are redundant.

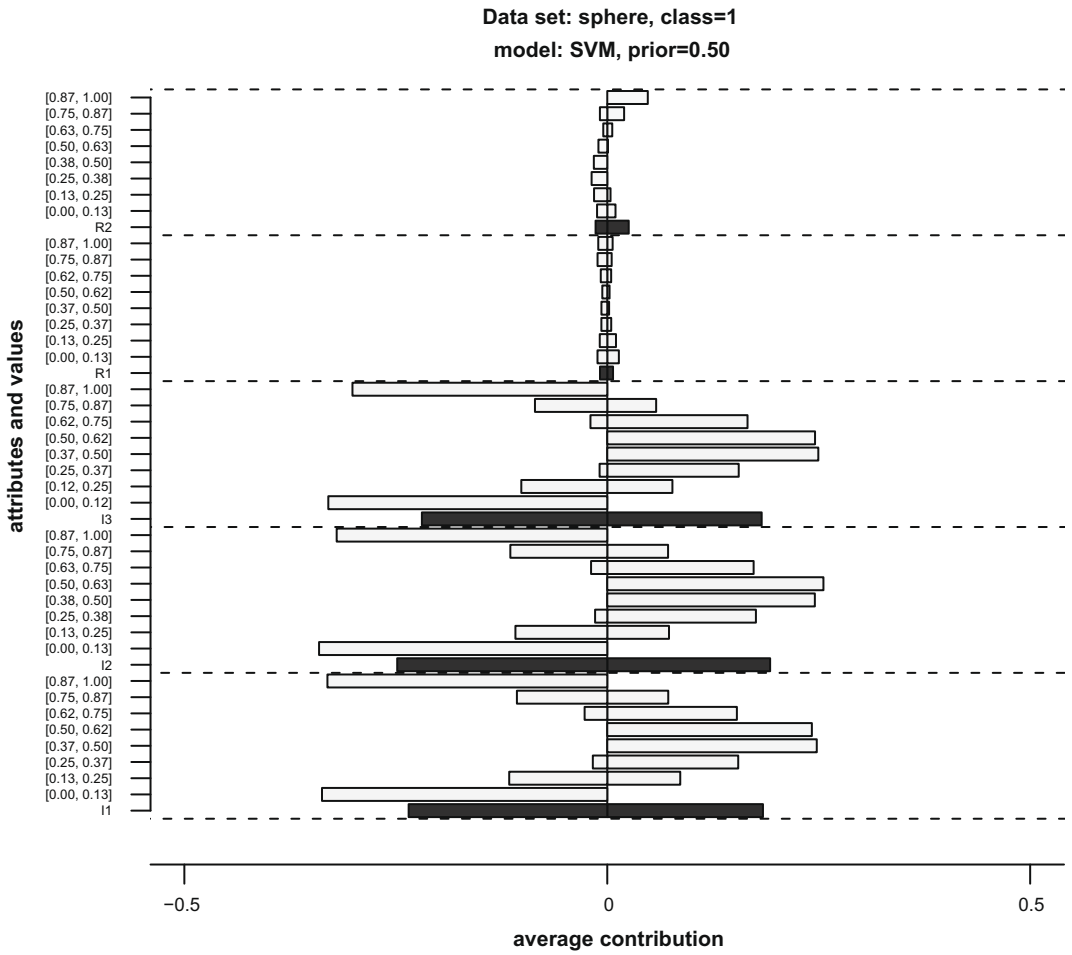


Fig. 4. Model explanation of the *sphere* data set using the *SVM* model. The dark bars represent the mean negative and mean positive contributions of individual features across all test instances. The light bars represent the mean negative and mean positive contributions of individual feature values across all test instances. Continuous features are discretized for visualization purposes and the mean contribution is used for each interval.

Our second example on Fig. 5b is an instance from the *xor* data set. We are explaining the decision of a decision tree, which performs very well on this data set and correctly classifies this instance. The explanation also correctly assigns an equal negative contribution to each important feature value. Because the concept behind this data set is the parity problem, the classifier can not make any conclusions based on any single feature value. The vertical lines correctly reflect this, as no feature value has a significant contribution on its own. Only when the values of the three important features are observed together, can a decision be made. By observing the list of interactions, we can see that the contribution of the interaction of the three important features is equal to the difference between the prior probability and the predicted probability. Only a single interaction is listed because no other interaction substantially contributes to the decision Therefore, the interaction between $I_1 = 1$, $I_2 = 0$, $I_3 = 1$, is the sole important contributor to the model's decision.

4.2. Applying the method to a real-life oncology data set

Breast cancer recurrence prediction is an important aspect of oncology. It helps to identify patients with the most critical prognoses and reduces the number of unnecessary therapies. Oncologists from the Institute of Oncology, Ljubljana, Slovenia have provided us with a breast cancer data set with 949 instances, which serves as an excellent test and an example of the usefulness of our explanation method. Each instance is an individual breast cancer patient and is described with 13 features recorded at the time of breast cancer surgery. The class is either 1 (recurrence within 5 years) or 2 (no recurrence within 5 years). A more detailed description of the features can be found in Table 4.

In Fig. 6, we can see the model explanation for the oncological data set using the *IME* method, and in Fig. 7 we can see the model explanation for the same data set generated using the Robnik-Šikonja and Kononenko method [20]. Both were generated for the Random forests algorithm [4], which was the best performing model on the data set. The model achieved an

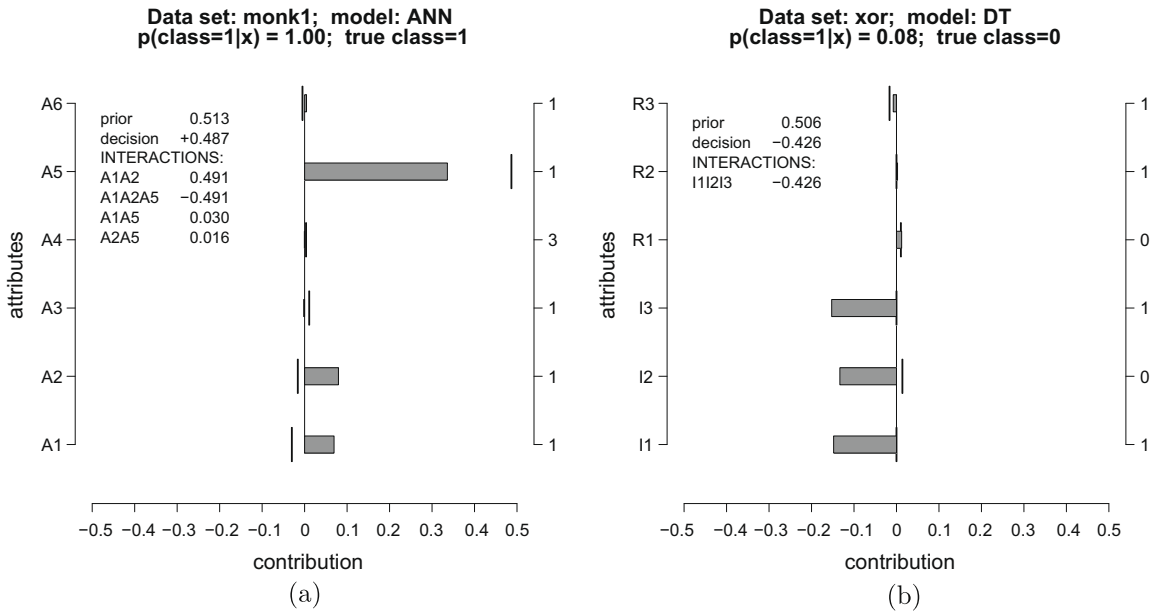


Fig. 5. Two enhanced *IME* instance explanations. A vertical line next to each bar indicates the feature values contribution on its own. A list of most important interaction contributions offers additional insight into possible interactions between feature values.

accuracy of 0.73 across 100 test instances, which was slightly better than the oncologists' accuracy on the same test instances. Note that both oncologists and the Random forests model are significantly better predictors than random predictions and the default predictor (i.e., always predicting the majority class value). The Random forests generated model uses more features, even features of minor significance, to produce a prediction. Subsequently, there are many small interactions between feature values, and the *EXPLAIN* method produces a model explanation that is difficult to interpret. On the other hand, *IME* computes all the interactions and divides them among the feature values that are part of the interactions. This leads to, as we can see in Fig. 6, a more informative explanation, with the contributions of individual values more clearly

Table 4
A detailed description of the features involved in breast cancer recurrence prediction and their values.

Feature name	Feature description
menop	Binary feature indicating menopausal status
stage	Tumor stage 1: less than 20 mm, 2: between 20 mm and 50 mm, 3: over 50 mm
grade	Tumor grade 1: good, 2: medium, 3: poor, 4: not applicable, 9: not determined
histType	Histological type of the tumor 1: ductal, 2: lobular, 3: other
PgR	Level of progesterone receptors in tumor (in fmol per mg of protein) 0: less than 10, 1: more than 10, 9: unknown
invasive	Invasiveness of the tumor 0: no, 1: invades the skin, 2: the mamilla, 3: skin and mamilla, 4: wall or muscle
nLymph	Number of involved lymph nodes 0: 0, 1: between 1 and 3, 2: between 4 and 9, 3: 10 or more
famHist	Medical history 0: no cancer, 1: 1st generation breast, ovarian or prostate cancer 2: 2nd generation breast, ovarian or prostate cancer, 3: unknown gynecological cancer 4: colon or pancreas cancer, 5: other or unknown cancers, 9: not determined
LVI	Binary feature indicating lymphatic or vascular invasion
ER	Level of oestrogen receptors in tumor (in fmol per mg of protein) 1: less than 5, 2: 5 to 10, 3: 10 to 30, 4: more than 30, 9: not determined
maxNode	Diameter of the largest removed lymph node 1: less than 15 mm, 2: between 15 and 20 mm, 3: more than 20 mm
posRatio	Ratio between involved and total lymph nodes removed 1: 0, 2: less that 10%, 3: between 10% and 30%, 4: over 30%
age	Patient age group 1: under 40, 2: 40–50, 3: 50–60, 4: 60–70, 5: over 70 years

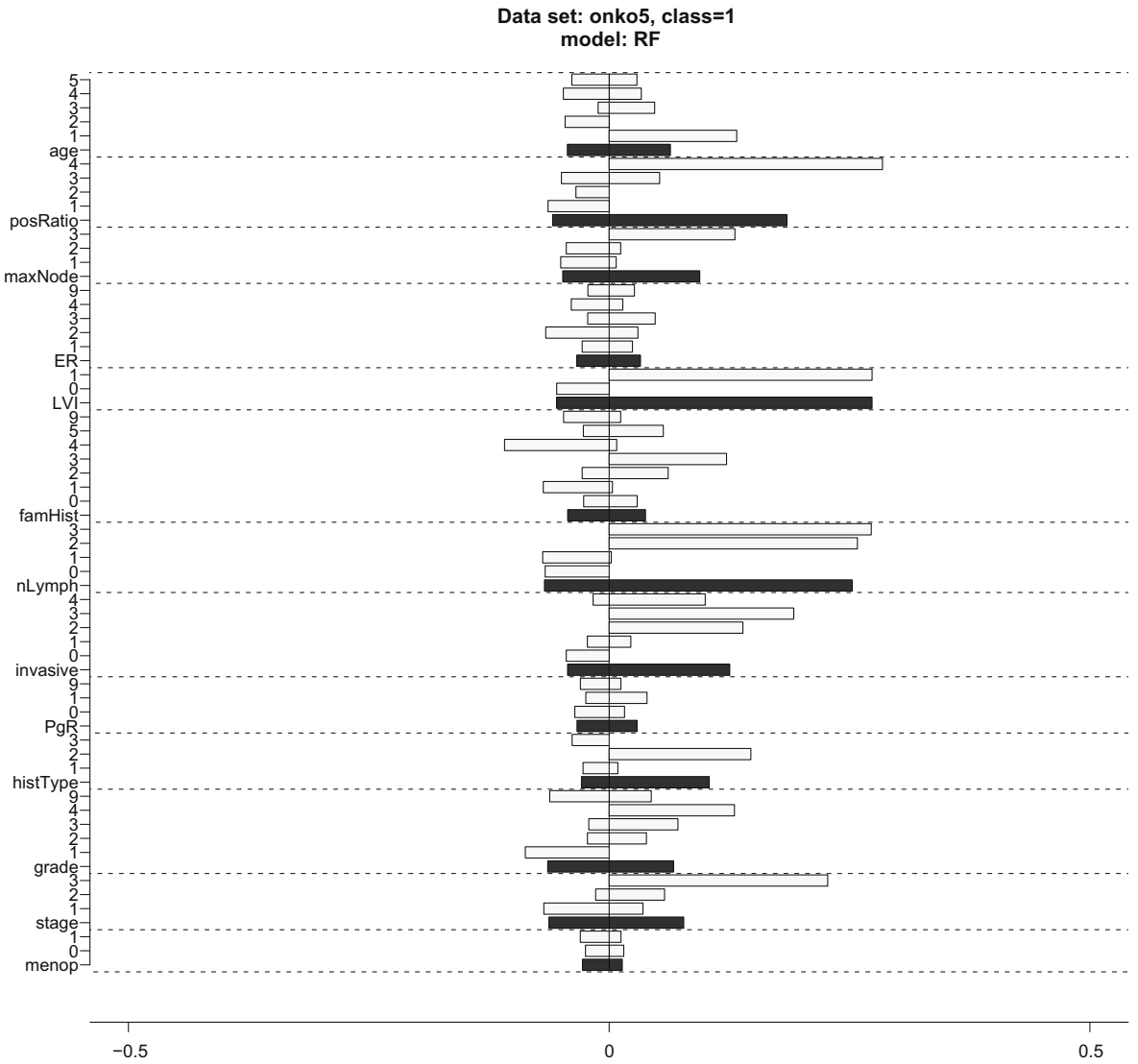


Fig. 6. Model explanation for the Random forests model and the oncology data set using our method.

expressed. Indeed, oncologists have confirmed that the model explanations reflect their expert medical knowledge regarding the importance of individual feature values in breast cancer recurrence prediction. An oncologist can use the model explanation to get an overview of how the model functions. For example, when observing Fig. 6 the expert would first look at the dark bars which reveal the overall influence of the feature on the model's decision. Features LVI, nLymph, and posRatio would be identified as most influential and menop, PgR, and ER as the least influential features. Then the oncologist can move on to individual feature values which reveal that LVI = 1 has a distinctly positive contribution (speaks in favor of a recurrence), low values of nLymph have a negative contribution, while high values of nLymph have a distinctly positive contribution, and so on. Because these conclusions are in agreement with current medical knowledge the expert oncologist can conclude that the model has learned knowledge that is relevant for breast cancer recurrence prediction.

To evaluate *IME* instance explanations, we generated 20 instance explanations and gave them to an expert oncologist for evaluation. The 20 instances were chosen from 100 test instances which were not included in the training set. To ensure diversity, the instances were selected in a semi-random way so that half of the instances had no recurrence and half had a recurrence. In each half, there were five correctly classified instances and five misclassified instances. In practice the outcome for a patient is not known in advance so it is important to include misclassifications because it enables us to observe whether the classifier is incorrect or if it has made a justifiable mistake. The latter means that the expert oncologist agrees with the explanation and would make the same incorrect prediction for that instance.

For each instance, we used the following evaluation procedure: For each feature value, the oncologist had to either agree with the generated contribution or disagree. To agree with the contribution, the oncologist had to agree with both the size of

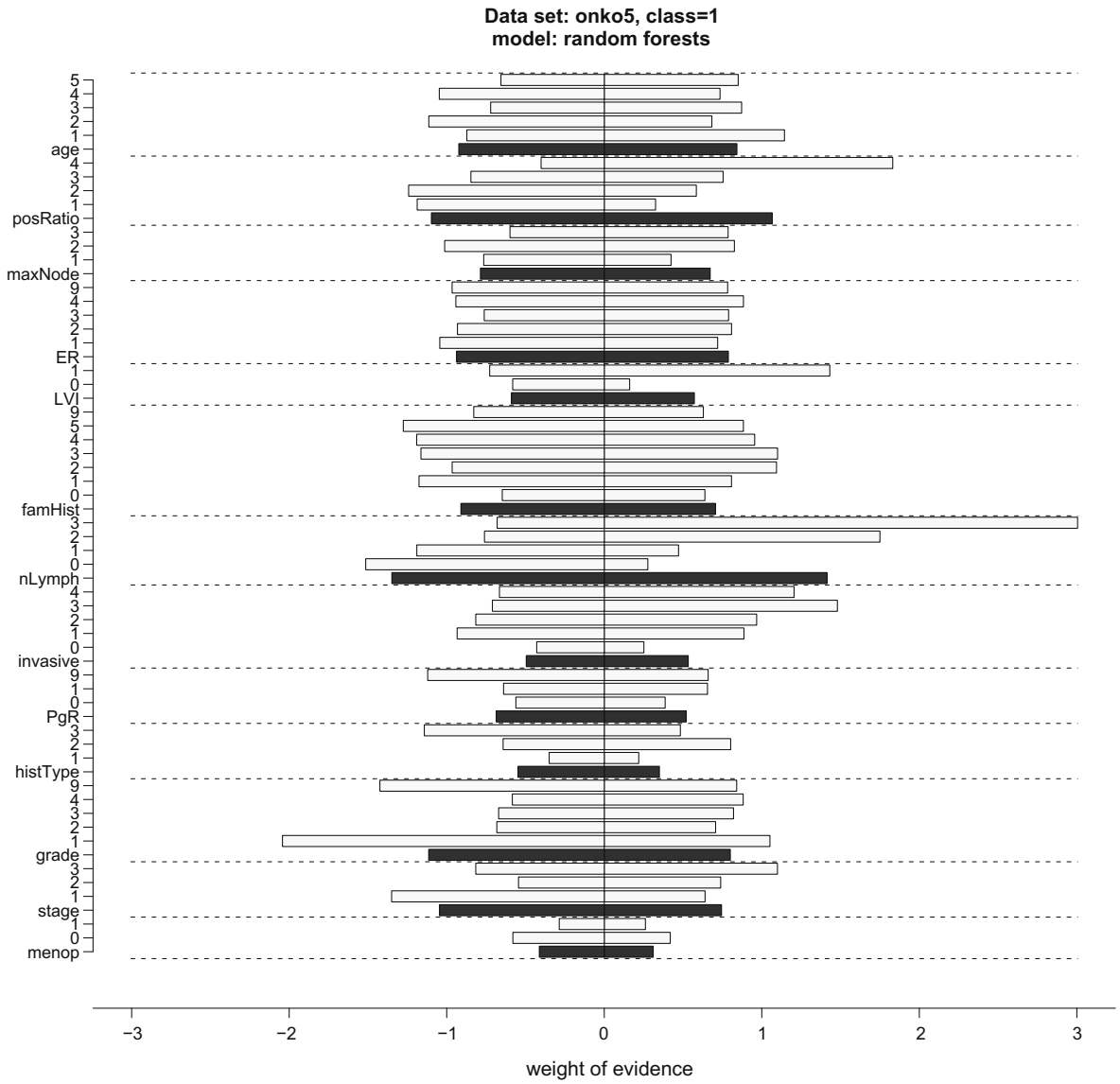


Fig. 7. Model explanation for the Random forests model and the oncology data set using the *EXPLAIN* method.

the contribution and the direction of the contribution. Note that out of the 13 features, the oncologists use only 9 in everyday medical practice. Therefore, the contributions of 4 features (histType, famHis, maxNode and posRatio) could not be evaluated, and our evaluation produced a total of 180 agreements/disagreements. The results are shown in Table 5, and we can see that there were a total of 21 disagreements, i.e., 12% of all contributions.

Eleven of the disagreements are on the features *age* and *menop*, which is also an age-related feature. Oncologists have established that these contributions in fact reflect the data and that their incorrectness is due to a bias in the data. As oncologists explained, a higher than usual number of young patients in our data were treated with therapies, because youth was, and still is, considered a negative factor. Consequently, young age sometimes moves the model's decision towards non-recurrence, because it implies a higher probability of therapy, which, in turn, reduces the chance of recurrence. This leaves us with 10 actual disagreements, which results in an encouraging 94% agreement rate with the contributions generated by our explanation.

Finally, in Fig. 8 we have two explanations for two different patients. We can use these explanations to examine how an expert oncologist or even a non-expert user can interpret the model's decision. On the left-hand side is an instance explanation for a breast cancer recurrence prediction using the Random forests model. The large number of positive lymph nodes (nLymph) and the large ratio between positive lymph nodes and total lymph nodes (posRatio) both speak heavily towards recurrence for this patient. The remaining feature values speak against recurrence but do not outweigh the two most

Table 5
The number and position of disagreements (×) for the 20 instances used for evaluation.

#	class	prediction	age	ER	LVI	nLymph	invasive	PgR	grade	stage	menop
1	2	2	✓	✓	✓	✓	✓	✓	✓	✓	×
2	2	1	✓	✓	✓	✓	✓	✓	✓	✓	✓
3	2	2	✓	✓	✓	✓	✓	✓	×	✓	✓
4	2	1	×	✓	✓	✓	✓	✓	✓	✓	×
5	2	1	✓	✓	✓	✓	✓	✓	✓	✓	✓
6	2	2	×	✓	✓	✓	✓	✓	✓	✓	×
7	2	1	×	×	✓	×	×	✓	✓	✓	✓
8	2	2	×	✓	✓	✓	✓	✓	✓	✓	✓
9	2	2	✓	✓	✓	✓	✓	✓	✓	✓	✓
10	2	1	✓	✓	✓	✓	✓	✓	×	✓	✓
11	1	1	✓	✓	✓	✓	✓	✓	✓	✓	✓
12	1	1	✓	×	✓	×	✓	✓	×	✓	✓
13	1	2	✓	✓	✓	✓	✓	✓	✓	✓	✓
14	1	2	×	✓	✓	✓	✓	✓	✓	✓	✓
15	1	1	✓	✓	✓	✓	✓	✓	✓	✓	✓
16	1	2	✓	✓	✓	✓	✓	✓	✓	✓	✓
17	1	2	×	✓	✓	✓	✓	×	×	✓	×
18	1	1	×	✓	✓	✓	✓	✓	✓	✓	✓
19	1	1	✓	✓	✓	✓	✓	✓	✓	✓	✓
20	1	2	✓	✓	✓	✓	✓	✓	✓	✓	✓
Sum			7	2	0	2	1	1	4	0	4

influential values. This results in the model's prediction that the probability of a recurrence is high (0.75). On the right-hand side is an explanation of the *NB* model's prediction for a different patient. The value stage = 3 (i.e., poor differentiation of the tumor) is the only value that notably speaks towards recurrence. However, three values have a considerable contribution against recurrence, so the model predicts a low probability of recurrence (0.06). These three values are: small tumor size (stage = 0), absence of positive lymph nodes (nLymph = 0), and subsequently lowest possible ratio of positive lymph nodes (posRatio = 0).

5. Discussion of time complexity

Admittedly, the exponential time complexity is the biggest obstacle barring wider applicability of *IME*. However, we have shown that the method can already be successfully applied to real-world problems, as is. Furthermore, the explanation

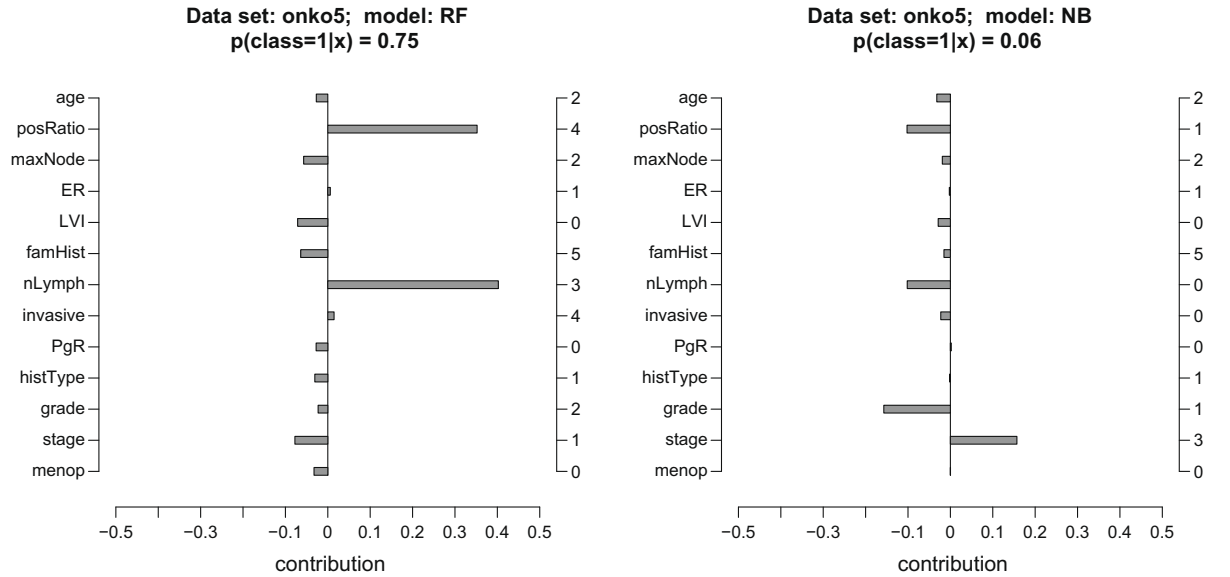


Fig. 8. Explanations of breast cancer recurrence predictions for two different patients.

method can be combined with feature selection methods, thus reducing the number of features and enabling us to use *IME* on more data sets. While both feature selection and explanation methods investigate the relevance of features, there are several important differences between the two. First, the main goal of feature selection is to select a feature subset that maximizes the classifier's performance. The goal of explanation methods is to explain the decisions of that classifier to the user, usually when it has been established that it is the best classifier, and it is already being used in practice. Therefore, feature selection and explanation are applied separately, at different stages of modeling. Second, feature selection deals with features at the model level, while *IME* deals with feature values on the instance level. And third, feature selection optimizes the prediction and is primarily interested in the effects of feature inclusion or exclusion on prediction quality. On the other hand, the goal of explanation methods is to reveal as many details about the influence and interactions of features as possible, which requires a higher time complexity. A brief analysis of the UCI machine learning repository [2] and existing feature selection publications [1,12,15] revealed that, of the 115 classification data sets (the total number of data sets in the repository is 174), most data sets either already have a small number of features (15 or less) or can be reduced to a small number of relevant features without compromising the prediction quality. This implies that for most data sets the optimal model uses only a smaller number of features and that it is feasible to use a more complex method such as *IME*. For reference, Table 6 shows the running times on artificial data sets used in this paper. The explanations for 432 test instances in the Monk1 data set were generated in under one second for each model. The explanations for 100 test cases from the real life oncology data set were generated in 35 s for the *NB* model and 241 s for the Random forests model. Note that a desktop computer (2.4 GHz CPU, 2 GB RAM, running Windows XP) was used for generating the explanations. The application code was a straightforward, non-optimized implementation of the method in Java (v1.6.0), and is available via email request. All the measurements are mean running times across 10 runs.

The purpose of the work described in this paper was to develop an explanation method that can handle all the possible concepts, regardless of the data set or classification model used. This is what distinguishes *IME* from existing model-independent methods, and this will provide a solid foundation with which to compare approximation methods. The tradeoff is a high time complexity. While the development of an approximation method is beyond the scope of this paper, we want to provide some ideas to illustrate that developing such a method is indeed possible. Note that the contribution of the ith feature value (5) can be expressed using only Δ-terms (see Appendix for a detailed proof). With this equation, we can avoid computing individual interactions, if we are only interested in the final contribution:

$$\pi_i = \sum_{W \subseteq \{1,2,\ldots,n\}-\{i\}} \frac{1}{n\binom{n-1}{n-|W|-1}} (\Delta_{W\cup\{i\}} - \Delta_W) \tag{6}$$

Eq. (6) is a sum of all the changes that the inclusion of the ith feature's value causes across all possible subsets of features. While these changes might not be normally distributed, they do have a finite variance and, following the central limit theorem, the sum of a sample of these changes would be normally distributed. This would justify using a sampling method to approximate a feature value's contribution. Approximating an individual interaction contribution is more challenging because interactions of all subsets are required if we want to compute an interaction of a set of feature values. Computing interaction contributions from the bottom up, only up to a certain size, would have a polynomial time complexity. This approach is partially justified by the fact that interactions of a larger number of features are less likely in real-world domains. Users

Table 6
Running times for generating an instance explanation for a single instance. Brackets indicate the running times for generating an explanation for all 1000 test instances at once and joining them into a model explanation.

	Running times (in seconds)				
	NB	*DT*	*kNN*	*SVM*	*ANN*
condInd	0.32	0.56	0.16	5.47	12.21
	(0.48)	(0.58)	(46.08)	(5.71)	(12.88)
xor	0.08	0.10	0.04	1.11	2.40
	(0.14)	(0.12)	(11.60)	(1.15)	(2.52)
group	0.02	0.01	0.01	0.51	0.62
	(0.02)	(0.01)	(4.22)	(0.60)	(0.69)
cross	0.08	0.06	0.04	0.75	2.28
	(0.12)	(0.07)	(19.50)	(0.88)	(2.52)
chess	0.02	0.01	0.01	0.19	0.51
	(0.02)	(0.01)	(4.22)	(0.21)	(0.58)
sphere	0.04	0.03	0.02	0.87	1.15
	(0.07)	(0.05)	(5.38)	(0.90)	(1.23)
disjunct	0.04	0.03	0.02	0.37	1.19
	(0.06)	(0.03)	(7.27)	(0.41)	(1.25)
random	0.02	0.01	0.01	0.20	0.51
	(0.03)	(0.01)	(4.34)	(0.24)	(0.69)

also benefit more from interactions of smaller subsets, because interactions of a larger number of features are more difficult to comprehend.

6. Conclusion

In this paper, we have proposed a new explanation method, *IME*, for explaining classifier decisions. The method explains a model's decision for an instance, and it can be used for any classifier. The instance explanation is provided in the form of feature value contributions, which describe how individual feature values contribute to the decision. Additionally, interaction contributions are provided, which describe how interactions between subsets of features contribute to the model's decision. Both feature value contributions and interaction contributions are expressed as a difference in probability, and their sum is implicitly normalized. This makes it easier to interpret the explanations and to compare how different models classify the same instance. Results on several artificial data sets and classification models show that *IME* explanations closely follow the quality of the model. An analysis of the explanations generated by *IME* showed that the contributions reveal the influence of the feature values and that the explanations provide insight into how the model learns from the data set. The application of *IME* to an oncological data set has shown not only that the method can already be applied in practice but also that expert oncologists agree with a vast majority of the generated explanations. We conclude that *IME* is a significant improvement over existing model-independent explanation methods and is especially useful when detailed explanations are needed.

The most important part of future work is developing an efficient approximation method. There are also some minor issues that need to be resolved, such as improving the visualization and providing a more detailed model explanation to the user. The use of *IME* for feature selection is also a possibility worth exploring. While the method's time complexity makes other feature selection methods more appropriate for optimizing prediction quality, the explanations generated by *IME* may still be used to identify specific feature values or interactions where the model behaves incorrectly. As a part of future work on this topic, we also want to explore the possibility of extending the method to regression models.

Appendix A. Proof of the contribution expression

The recursive definition of an interaction contribution (4) can be transformed into a non-recursive form. The non-recursive equation is identical to the inclusion/exclusion principle in set theory when looking for the intersection size (also being the Möbius inversion [21] of the set of all subsets, partially ordered by inclusion):

$$\mathscr{I}_Q = \sum_{W \subseteq Q} ((-1)^{|Q|-|W|} \Delta_W) \tag{7}$$

This equation is similar to multivariate mutual information in information theory, the difference being that marginal predictions are used instead of entropy. There are publications that do not directly deal with explanation but are highly related to our work. Several approaches to multivariate mutual information have been taken by Bell [3], Han [9], Yeung [25] and Jakulin [10]. The last of these also includes further references to interaction-related publications and is most suitable for those who want to become familiar with the subject of interactions from an information-theoretic view.

The non-recursive definition of an interaction contribution can be used to express the contribution of an arbitrary element of the set. We will start by using the definition of a feature value's contribution (5) and writing the contribution of a single feature from the set of n features. We can safely assume that the feature in question is labeled with 1:

$$\pi_1 = \sum_{W \subseteq \{2,\dots,n\}} \frac{\mathscr{I}_{W\cup\{1\}}}{|W|+1} \tag{8}$$

Let us examine the number of appearances of Δ_Q on the right-hand side of (8) after we expand each interaction using (4). Let N_{Δ_Q} be the sum of all such appearances, n the total number of elements, and $k = |\{2,\dots,n\} - Q|$. First, we will consider N_{Δ_Q}, where $1 \in Q$. The term Δ_Q appears only in interactions I_W where $Q \subseteq W$ and only once in each such interaction. The smallest such set W is, of course, Q, where Δ_Q appears with a positive sign. In interactions I_R, where $|R| = |Q| + 1$, it features with a negative sign and there are exactly $\binom{k}{1}$ such interactions in the sum in (8), because we can choose the additional element from the remaining k elements that are not already in the set $|Q|$. If we write all such terms up to I_W, where $|W| = n$, and take into account that each interaction I_W is divided by $|W|$, we get the series:

$$N_{\Delta_Q} = \frac{\binom{k}{0}}{n-k} - \frac{\binom{k}{1}}{n-k+1} + \cdots + (-1)^k \frac{\binom{k}{k}}{n}, \quad \text{where } 1 \in Q \tag{9}$$

N_{Δ_Q} is similar for $1 \notin Q$. The only difference is, that the smallest such set $W$where Δ_Q appears in I_W is of size $|Q| + 1$, because 1 is always in W for each interaction that features on the right-hand side of (8). Because of this, the difference becomes odd, and the first term in the series starts with a negative sign:

$$N_{\Delta_Q} = -\frac{\binom{k}{0}}{n-k} + \frac{\binom{k}{1}}{n-k+1} + \cdots + (-1)^{k+1}\frac{\binom{k}{k}}{n}, \quad \text{where } 1 \notin Q \tag{10}$$

Let us name the series that appear in (9) and (10) as $V(n,k)$ and $-V(n,k)$, respectively. Now, let us analyze the series $V(n,k)$. We start by taking the binomial $(1-x)^k$ and expanding it. We then multiply both sides with x^{n-k-1} and integrate them, which transforms the right side of the equation into the series $V(n,k)$ and enables us to express the left side with beta function $B(p,q)$:

$$\begin{aligned}
(1-x)^k &= \binom{k}{0} - \binom{k}{1}x + \binom{k}{1}x^2 - \cdots \pm \binom{k}{k}x^k \\
x^{n-k-1}(1-x)^k &= \binom{k}{0}x^{n-k-1} - \binom{k}{1}x^{n-k} + \cdots \pm \binom{k}{n-1}x^k \\
\int_0^1 x^{n-k-1}(1-x)^k dx &= \int_0^1 \left(\binom{k}{0}x^{n-k-1} - \binom{k}{1}x^{n-k} + \cdots \pm \binom{k}{n-1}x^k\right) dx \\
B(n-k,k+1) &= \frac{\binom{k}{0}}{n-k} - \frac{\binom{k}{1}}{n-k+1} + \cdots \pm \frac{\binom{k}{k}}{n} = V(n,k)
\end{aligned}$$

Using the known relation between beta and gamma function $B(p,q) = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)}$ and the definition of gamma function $\Gamma(a) = (a-1)!$, we can further derive our series:

$$V(n,k) = B(n-k,k+1) = \frac{\Gamma(n-k)\Gamma(k+1)}{\Gamma(n+1)} = \frac{(n-k-1)!k!}{n!} = \frac{1}{n\binom{n-1}{k}}$$

Let $S = \{1,2,\ldots,n\}$. Now we are ready to derive the desired contribution:

$$\begin{aligned}
\pi_i &= \sum_{W \subseteq S-\{i\}} V(n, |S-\{i\}-W|)\Delta_{W\cup\{i\}} - \sum_{W \subseteq S-\{i\}} V(n, |S-\{i\}-W|)\Delta_W \\
&= \sum_{W \subseteq S-\{i\}} V(n, n-|W|-1)(\Delta_{W\cup\{i\}} - \Delta_W) \\
&= \sum_{W \subseteq S-\{i\}} \frac{1}{n\binom{n-1}{n-|W|-1}}(\Delta_{W\cup\{i\}} - \Delta_W)
\end{aligned} \tag{11}$$

Appendix B. The Δ-terms of the true explanation of the sphere data set

Let an instance $A_1 = x_1, A_2 = x_2, A_3 = x_3, A_4 = x_4, A_5 = x_5$ represent a point $X(x_1,x_2,x_3)$. Note that A_4 and A_5 are unrelated to the class and thus irrelevant. To simplify the true explanation of the instance x we move the sphere center to the center of the coordinate system and scale the sphere radius to 1. Now the true contribution of an feature value depends only on it's absolute value and we can focus on a single quadrant. Each instance $X = (x_1,x_2,x_3)$ is transformed into $Y = (y_1,y_2,y_3)$:

$$y_i = 2 \cdot |x_i - \frac{1}{2}|; \quad i \in \{1,2,3\}$$

By dividing an eight of the sphere's volume by the volume of the unit cube, we derive that the prior probability of class value 1 is $\frac{\pi}{6}$. When a single coordinate is known, the other two coordinates define a plane. The intersection of that plane and the sphere quadrant is a quarter of a circle and the single feature Δ-terms equal that area (i.e., the probability of class value 1 if one coordinate is known) minus the prior probability:

$$\Delta_{\{i\}} = \frac{\pi}{4}(1-y_i^2) - \frac{\pi}{6}; \quad i \in \{1,2,3\}$$

When two coordinates are known, the remaining coordinate defines a line. The length of the intersection of that line and the sphere quadrant defines the two-feature Δ-terms. Two extreme coordinates are enough to put the entire instance out of the sphere and in that case the Δ-term is automatically 0 minus the prior probability:

$$\Delta_{\{i,j\}} = \begin{cases} \sqrt{1-(y_i^2+y_j^2)} - \frac{\pi}{6}; & y_i^2 + y_j^2 \leqslant 1; \\ 0 - \frac{\pi}{6}; & y_i^2 + y_j^2 > 1; \end{cases} \quad i,j \in \{1,2,3\}, \quad i \neq j$$

When all three coordinates are known, we know exactly where the instance lies and whether it is inside the sphere:

$$\Delta_{\{1,2,3\}} = \begin{cases} 1 - \frac{\pi}{6}; & y_1^2 + y_2^2 + y_3^2 \leqslant 1 \\ 0 - \frac{\pi}{6}; & y_1^2 + y_2^2 + y_3^2 > 1 \end{cases}$$

References

[1] A.E. Akadi, A.E. Ouardighi, A. Driss, A powerful feature selection approach based on mutual information, International Journal of Computer Science and Network Security 8 (2008) 116–121.
[2] A. Asuncion, D. Newman, UCI machine learning repository. <http://archive.ics.uci.edu/ml/>, 2008.
[3] A.J. Bell, The co-information lattice, in: Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation, Nara, Japan, 2003, pp. 921–926.
[4] L. Breiman, Random forests, Machine Learning Journal 45 (2001) 5–32.
[5] G.W. Brier, Verification of forecasts expressed in terms of probability, Monthly Weather Review 75 (1950) 1–3.
[6] J.R. Cano, F. Herrera, M. Lozano, Evolutionary stratified training set selection for extracting classification rules with trade off precision-interpretability, Data and Knowledge Engineering 60 (1) (2007) 90–108.
[7] A.L. de Santana, C. Frances, C.A. Rocha, S.V. Carvalho, N.L. Vijaykumar, L.P. Rego, J.C. Costa, Strategies for improving the modeling and interpretability of Bayesian networks, Data and Knowledge Engineering 63 (1) (2007) 91–107.
[8] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1157–1182.
[9] T.S. Han, Multiple mutual informations and multiple interactions in frequency data, Information and Control 46 (1) (1980) 26–45.
[10] A. Jakulin, Machine learning based on attribute interactions, Ph.D. Thesis, University of Ljubljana, Faculty of Computer and Information Science, Ljubljana. <http://eprints.fri.uni-lj.si/archive/00000205/01/jakulin05phd.pdf>, 2005.
[11] A. Jakulin, M. Možina, J. Demšar, I. Bratko, B. Zupan, Nomograms for visualizing support vector machines, in: KDD'05: Proceeding of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, ACM, New York, NY, USA, 2005, pp. 108–117.
[12] R. Kohavi, G. John, Wrappers for feature subset selection, Artificial Intelligence journal 97 (1–2) (1997) 273–324.
[13] I. Kononenko, Machine learning for medical diagnosis: history, state of the art and perspective, Artificial Intelligence in Medicine 23 (2001) 89–109.
[14] I. Kononenko, M. Kukar, Machine Learning and Data Mining: Introduction to Principles and Algorithms, Horwood Publ., 2007.
[15] Y. Lei, L. Huan, Feature selection for high-dimensional data: a fast correlation-based filter solution, in: Proceedings of the Twentieth International Conference on Machine Leaning (ICML-03), Washington, DC, 2003, pp. 856–863.
[16] V. Lemaire, R. Fraud, N. Voisine, Contact personalization using a score understanding method, in: International Joint Conference on Neural Networks (IJCNN), 2008.
[17] H. Liu, L. Yu, Toward integrating feature selection algorithms for classification and clustering, Knowledge and Data Engineering 17 (4) (2005) 491–502.
[18] J. Lubsen, J. Pool, E. van der Does, A practical device for the application of a diagnostic or prognostic function, Methods of Information in Medicine 17 (1978) 127–129.
[19] M. Možina, J. Demšar, M. Kattan, B. Zupan, Nomograms for visualization of naive bayesian classifier, in: PKDD '04: Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases, Springer-Verlag New York, Inc., New York, NY, USA, 2004, pp. 337–348.
[20] M. Robnik-Šikonja, I. Kononenko, Explaining classifications for individual instances, IEEE Transactions on Knowledge and Data Engineering 20 (2008) 589–600.
[21] R.P. Stanley, Enumerative Combinatorics, vol. 1–2, Cambridge University Press, 1999.
[22] D. Szafron, B. Poulin, R. Eisner, P. Lu, R. Greiner, D. Wishart, A. Fyshe, B. Pearcy, C. Macdonell, J. Anvik, Visual explanation of evidence in additive classifiers, in: Proceedings of Innovative Applications of Artificial Intelligence, 2006.
[23] S. Thrun, J. Bala, E. Bloedorn, I. Bratko, B. Cestnik, J. Cheng, K.D. Jong, S. Dzeroski, R. Hamann, K. Kaufman, S. Keller, I. Kononenko, J. Kreuziger, R. Michalski, T. Mitchell, P. Pachowicz, B. Roger, H. Vafaie, W.V. de Velde, W. Wenzel, J. Wnek, J. Zhang, The MONK's problems: a performance comparison of different learning algorithms, Tech. Rep. CMU-CS-91-197, Carnegie Mellon University, Computer Science Department, Pittsburgh, PA, 1991.
[24] G. Towell, J.W. Shavlik, Extracting refined rules from knowledge-based neural networks, machine learning, Machine Learning 13 (1993) 71–101.
[25] R.W. Yeung, A new outlook of Shannon's information measures, IEEE Transactions on Information Theory 37 (3) (1991) 466–474.

Erik Strumbelj is a Ph.D. student and junior researcher at the University of Ljubljana, Faculty of Computer and Information Science. His research interests include artificial intelligence, machine learning, data mining and forecasting.

Igor Kononenko received his Ph.D. in 1990 from the University of Ljubljana, Slovenia. He is a professor at the Faculty of Computer and Information Science in Ljubljana and the head of the Laboratory for Cognitive Modeling. His research interests include artificial intelligence, machine learning, neural networks and cognitive modeling. He is a member of the editorial board of Applied Intelligence Journal and Informatica Journal. He is a (co)author of 190 papers and 10 textbooks. Recently he co-authored the book "Machine Learning and Data Mining: Introduction to Principles and Algorithms" (Horwood, 2007).

Marko Robnik-Sikonja received his Ph.D. in computer science in 2001 from the University of Ljubljana. He is an Assistant Professor at the University of Ljubljana, Faculty of Computer and Information Science. His research interest include machine learning, data mining, knowledge discovery, cognitive modeling, and practical applications. He is a (co)author of more than 40 publications in journals and international conferences.