Towards Interpretation of Pairwise Learning

Mengdi Huai,[1] **Di Wang,**[2] **Chenglin Miao,**[2] **Aidong Zhang**[1]
[1]Department of Computer Science, University of Virginia
[2]Department of Computer Science and Engineering, State University of New York at Buffalo
[1]{mh6ck, aidong}@virginia.edu, [2]{dwang45, cmiao}@buffalo.edu

Abstract

Recently, there are increasingly more attentions paid to an important family of learning problems called pairwise learning, in which the associated loss functions depend on pairs of instances. Despite the tremendous success of pairwise learning in many real-world applications, the lack of transparency behind the learned pairwise models makes it difficult for users to understand how particular decisions are made by these models, which further impedes users from trusting the predicted results. To tackle this problem, in this paper, we study feature importance scoring as a specific approach to the problem of interpreting the predictions of black-box pairwise models. Specifically, we first propose a novel adaptive Shapley-value-based interpretation method, based on which a vector of importance scores associated with the underlying features of a testing instance pair can be adaptively calculated with the consideration of feature correlations, and these scores can be used to indicate which features make key contributions to the final prediction. Considering that Shapley-value-based methods are usually computationally challenging, we further propose a novel robust approximation interpretation method for pairwise models. This method is not only much more efficient but also robust to data noise. To the best of our knowledge, we are the first to investigate how to enable interpretation in pairwise learning. Theoretical analysis and extensive experiments demonstrate the effectiveness of the proposed methods.

Introduction

In recent years, there has been increasing interest in an important family of learning problems that is categorized as pairwise learning (Boissier et al. 2016). Different from the traditional pointwise learning (e.g., regression and classification) (Yao et al. 2018) where the loss function takes only individual instances as its input, pairwise learning involves pairs of instances as the input of its loss function. Comparing to pointwise learning, pairwise learning is more capable of modeling the relative relationship between pairs of instances, which has been demonstrated in many real-world applications. For example, in patient similarity learning, the learner (e.g., a doctor/hospital) can learn a clinically meaningful similarity metric to measure the proximity between a pair of patients through formulating the learning task as a pairwise learning problem (Huai et al. 2018a). Additionally, many other learning tasks can also be classified as pairwise learning, such as AUC maximization (Ying, Wen, and Lyu 2016; Natole, Ying, and Lyu 2018), metric learning (Huai et al. 2019; Suo et al. 2018; Huai et al. 2018b), bipartite ranking (Tang and Wang 2018).

Despite its tremendous success in many real-world applications, pairwise learning still faces one challenging problem, i.e., the lack of transparency behind its behaviors, which makes it difficult for users to understand how particular decisions are made by the learned pairwise model. For instance, in the patient similarity learning task, the similarity metric is usually learned from a large amount of high dimensional and complex patient data. The learner can obtain the proximity between a pair of patients based on the learned metric, but he/she has no idea why the metric reports such proximity. The "black box" nature of the learned pairwise models may impede users from trusting the predicted results, especially when the model is used for making critical decisions (e.g., medical diagnosis), because the consequences may be catastrophic if the predictions are acted upon blind faith. The lack of transparency behind pairwise learning models has hampered their further applications in real world. Thus, it is essential to investigate how to enable interpretaion in pairwise learning.

In this paper, we aim to study feature importance scoring as a specific approach to the problem of interpreting the predictions of black-box pairwise models. Specifically, given a learned pairwise model and a testing instance pair, we hope to design an interpretation method that can generate a vector of importance scores associated with the underlying features of the testing instance pair, and enable these importance scores to indicate which features make key contributions to the final predicted result. There is now many interpretation methods that can score the importance of the input features for traditional pointwise learning models (e.g., classification models). Among them, the Shapley-value-based methods (Ribeiro, Singh, and Guestrin 2016; Datta, Sen, and Zick 2016; Shrikumar, Greenside, and Kundaje 2017; Štrumbelj and Kononenko 2014; Lundberg and Lee 2017; Chen et al. 2018; Ancona, Öztireli, and Gross 2019; Kononenko and others 2010) have drawn significant attention as they are the only methods that can provide theoretical guarantee. However, these methods cannot be directly used for pairwise models. First of all, to score the impor-

tance of a subset of input features, these methods usually need to pre-define a reference vector to mask the rest features. An implicit assumption in these methods is that all the testing instances use the same reference vector, which is unreasonable for pairwise models. When interpreting the predictions made by pairwise models, if both instances in the testing pair use the same reference vector, the relationship between them will be largely affected (e.g., may make them more similar) and wrong prediction may be generated. Additionally, existing interpretation methods for pointwise learning usually assume that the input features are nearly independent. However, in practice, the features may be correlated with each other and the correlation can also affect the predictions made by the models (Xie et al. 2018).

To address the above challenges, in this paper, we first propose a novel adaptive Shapley-value-based interpretation method for pairwise models (**ASIPair**), which not only takes into account feature correlations but also can adaptively calculate the importance scores of the underlying features for each testing instance pair. We also provide theoretical analysis to show that the proposed adaptive method is the unique solution with the desired properties. Considering that Shapley-value-based methods are usually computationally challenging, we further propose a robust approximation interpretation method for pairwise models (**RAIPair**), which is motivated by the fact that not all features are important and only a subset of features contain the discriminative information for the final predicted result. The proposed approximated interpretation method does not make any assumptions on the underlying feature structure and is also robust to data noise. To the best of our knowledge, we are the first to investigate how to enable interpretation in pairwise learning. Both theoretical analysis and extensive experiments demonstrate the effectiveness of the proposed interpretation methods for pairwise learning.

Problem Definition

Let $\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i)$ be a random instance drawn from an unknown distribution $\mathcal{P}$ on $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{Y}}$, where $\boldsymbol{x}_i \in \boldsymbol{\mathcal{X}} \subset \mathbb{R}^D$ represents the D-dimensional feature vector and $y_i \in \boldsymbol{\mathcal{Y}} \subset \mathbb{R}$ represents the class label. Pairwise learning refers to the learning tasks where the associated loss function involves a pair of instances. Specifically, for any two instances $\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i)$ and $\boldsymbol{z}_j = (\boldsymbol{x}_j, y_j)$, the loss function for pairwise learning usually takes the form $V(\zeta, (\boldsymbol{x}_i, y_i), (\boldsymbol{x}_j, y_j))$ for a hypothesis function $\zeta : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$. Many learning tasks, such as metric learning, AUC maximization and ranking, can be cast into the framework of pairwise learning. For example, the Mahalanobis-based metric learning aims to learn a Mahalanobis distance function $\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)'\boldsymbol{W}(\boldsymbol{x}_i - \boldsymbol{x}_j)$ encoded by a semi-positive matrix $\boldsymbol{W} \in \mathbb{S}^{D\times D}$ to bring instances with similar labels together while keeping instances with different labels apart. With the learned function ζ, we can calculate the similarity degree of a test instance pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. A choice of the loss function in metric learning is the logistic loss (Huai et al. 2018a), i.e., $V(\zeta, (\boldsymbol{x}_i, y_i), (\boldsymbol{x}_j, y_j)) = \log(1 + \exp(-y_i y_j(\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j) - 1))$. Then the distance function ζ can be learned through minimizing the below expected risk

$$\mathcal{R}(\zeta) = \int\int_{\boldsymbol{\mathcal{Z}}\times\boldsymbol{\mathcal{Z}}} V(\zeta, (\boldsymbol{x}_i, y_i), (\boldsymbol{x}_j, y_j)) d\mathcal{P}(\boldsymbol{x}_i, y_i) d\mathcal{P}(\boldsymbol{x}_j, y_j). \quad (1)$$

Note that $V(\cdot)$ can also be other loss functions, such as the exponential loss and hinge loss.

Our goal in this paper is to develop a general interpretation method that can provide explanations for the results predicted by pairwise models. Specifically, given a testing instance pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and a learned pairwise model, the pairwise model will make a prediction (e.g., similar degree in metric learning task) for $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. We aim to illustrate why such a prediction is made through identifying a set of important features in $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ that make key contributions to the predicted result.

Methodology

In this section, we describe the proposed interpretation methods for pairwise models. Specifically, we first propose an adaptive Shapley-value-based interpretation method (called ASIPair) with the consideration of feature correlations. Considering that Shapley-value-based methods are usually computationally challenging, we then propose a robust approximation interpretation method (called RAIPair).

Adaptive Interpretation Method for Pairwise Models

The importance of each feature in $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ can be reflected by its contribution to the final predicted result. For any given subset $T \subset [D] = \{1, 2, ..., D\}$, we use $\boldsymbol{x}_i^T = \{x_{i,t}, t \in T\}$ to denote the associated sub-vector of features, where $x_{i,t}$ denotes the t-th element in $\boldsymbol{x}_i$. Let $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T]$ be the induced expected conditional prediction for the testing instance pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ when it is restricted to using only the sub-vectors $\boldsymbol{x}_i^T$ and $\boldsymbol{x}_j^T$. Then, for a given subset $T \subset [D] \setminus \{d\}$, the marginal contribution of the d-th feature to T (joining the subset T) can be calculated as follows

$$\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta) = \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d\}}, \boldsymbol{x}_j^{T\cup\{d\}}] - \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T]. \quad (2)$$

To obtain $\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta)$, we need to calculate the two expected pairwise conditional functions $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d\}}, \boldsymbol{x}_j^{T\cup\{d\}}]$ and $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T]$.

As described in the introduction section, existing interpretation methods developed for traditional pointwise learning models cannot be directly used here to calculate $\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta)$ for pairwise learning. There is an implicit assumption in these methods that all the testing instances use the same pre-defined reference vector that is used for replacing $x_{i,t}$ ($t \in [D]\setminus T$) when measuring the contribution of $\boldsymbol{x}_i^T$ to the prediction, which is unreasonable for pairwise models. For pairwise models, if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ use the same reference vector, we cannot obtain reasonable $\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta)$, because the relationship between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ will be largely affected and wrong prediction may be generated. Furthermore,

these methods assume that the input features are nearly independent. However, in practice, the features are usually correlated with each other and the correlations can also affect the predicted result. To address the above challenges, we propose the following calculation method for $\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta)$.

Since the calculation procedures for $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d\}}, \boldsymbol{x}_j^{T\cup\{d\}}]$ and $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}]$ are similar, here we take $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}]$ as an example to describe the calculation procedure. We first rewrite the expected conditional pairwise function $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}]$ as follows

$$\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}] = \int\int \zeta([\boldsymbol{x}_i^{\bar{T}}, \boldsymbol{x}_i^{T}], [\boldsymbol{x}_j^{\bar{T}}, \boldsymbol{x}_j^{T}]) P(\boldsymbol{x}_i^{\bar{T}}|\boldsymbol{x}_i^{T}) P(\boldsymbol{x}_j^{\bar{T}}|\boldsymbol{x}_j^{T}) d\boldsymbol{x}_i^{\bar{T}} d\boldsymbol{x}_j^{\bar{T}}, \quad (3)$$

where $\bar{T} = [D] \setminus T$ and $P(\boldsymbol{x}_i^{\bar{T}}|\boldsymbol{x}_i^{T})$ denotes the conditional distribution of $\boldsymbol{x}_i^{\bar{T}}$ given $\boldsymbol{x}_i^{T}$. $[\boldsymbol{x}_i^{\bar{T}}, \boldsymbol{x}_i^{T}]$ denotes the concatenation of $\boldsymbol{x}_i^{\bar{T}}$ and $\boldsymbol{x}_i^{T}$, i.e., $\boldsymbol{x}_i = [\boldsymbol{x}_i^{\bar{T}}, \boldsymbol{x}_i^{T}]$. To take the feature correlation into account, we propose to incorporate the covariance matrix that contains features' correlation information into the calculation process of the expected conditional pairwise function. Suppose the training set is denoted as $\{\boldsymbol{x}_k\}_{k=1}^K$, where K is the size of the training set. $\boldsymbol{x}_i^T$ and $\boldsymbol{x}_j^T$ can be transformed as

$$\tilde{\boldsymbol{x}}_i^T = \boldsymbol{\Omega}_T^{-1/2}(\boldsymbol{x}_i^T - \boldsymbol{\mu}_T), \quad \tilde{\boldsymbol{x}}_j^T = \boldsymbol{\Omega}_T^{-1/2}(\boldsymbol{x}_j^T - \boldsymbol{\mu}_T), \quad (4)$$

where $\boldsymbol{\mu}_T$ and $\boldsymbol{\Omega}_T$ denote the mean vector and covariance matrix of the set of sub-vectors $\{\boldsymbol{x}_k^T\}_{k=1}^K$ for training instances, respectively. The t-th element in $\boldsymbol{\mu}_T$ represents the mean value of the t-th feature over $\{\boldsymbol{x}_k^T\}_{k=1}^K$. Considering the fact that the training instance $\boldsymbol{x}_k = [\boldsymbol{x}_k^{\bar{T}}, \boldsymbol{x}_k^T]$ with $\boldsymbol{x}_k^T$ close to $\boldsymbol{x}_i^T$ is more informative when calculating $P(\boldsymbol{x}_i^{\bar{T}}|\boldsymbol{x}_i^T)$, we then propose to use the training instances $\{\boldsymbol{x}_k\}_{k=1}^K$ to empirically calculate the pairwise conditional expectation $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T]$. Specifically, we first calculate the distance between $\boldsymbol{x}_i$ and each instance $\boldsymbol{x}_k$ in the training set $\{\boldsymbol{x}_k\}_{k=1}^K$ as

$$Q_i^T(\boldsymbol{x}_i, \boldsymbol{x}_k) = (\tilde{\boldsymbol{x}}_i^T - \boldsymbol{\Omega}_T^{-1/2}(\boldsymbol{x}_k^T - \boldsymbol{\mu}_T))'(\tilde{\boldsymbol{x}}_i^T - \boldsymbol{\Omega}_T^{-1/2}(\boldsymbol{x}_k^T - \boldsymbol{\mu}_T))/|T|. \quad (5)$$

The distance between the testing instance $\boldsymbol{x}_j$ and the training instance $\boldsymbol{x}_k$ can be calculated in a similar way. Then, for each pair $(\boldsymbol{x}_i, \boldsymbol{x}_k)$ where $k \in [K]$, we calculate a weight $w_T(\boldsymbol{x}_i, \boldsymbol{x}_k) = \exp(-Q_i^T(\boldsymbol{x}_i, \boldsymbol{x}_k)/2\sigma^2)$, where σ is a smoothing parameter (the value is set as 0.2 in our experiment). After deriving all the weights $\{w_T(\boldsymbol{x}_i, \boldsymbol{x}_k)\}_{k=1}^K$, we sort these weights in an increasing order, and we use $\boldsymbol{x}_{k'}$ to denote the training instance corresponding to the k'-th element in the ordered weight set. Similarly, we can derive the weights $\{w_T(\boldsymbol{x}_j, \boldsymbol{x}_k)\}_{k=1}^K$ for $\boldsymbol{x}_j$ and order them in an increasing order. Let $\boldsymbol{x}_{k''}$ be the training instance corresponding to the k''-th element in the ordered weight set for $\boldsymbol{x}_j$. Then, we can estimate $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T]$ as

$$\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^T, \boldsymbol{x}_j^T)] = \frac{1}{\sum_{k'=1}^{K_1} w_T(\boldsymbol{x}_i, \boldsymbol{x}_{k'}) \sum_{k''=1}^{K_1} w_T(\boldsymbol{x}_j, \boldsymbol{x}_{k''})} \{\sum_{k'=1}^{K_1} w_T(\boldsymbol{x}_i, \boldsymbol{x}_{k'}) \cdot [\sum_{k''=1}^{K_1} w_T(\boldsymbol{x}_j, \boldsymbol{x}_{k''}) \zeta([\boldsymbol{x}_{k'}^{\bar{T}}, \boldsymbol{x}_i^T], [\boldsymbol{x}_{k''}^{\bar{T}}, \boldsymbol{x}_j^T])]\}, \quad (6)$$

where K_1 denotes the number of the selected training instances, and it can be decided as

$$K_1 = \arg\min_{L\in[K]} \{\frac{\sum_{k'=1}^{L} w_T(\boldsymbol{x}_i, \boldsymbol{x}_{k'}) \sum_{k''=1}^{L} w_T(\boldsymbol{x}_j, \boldsymbol{x}_{k''})}{\sum_{k'=1}^{K} w_T(\boldsymbol{x}_i, \boldsymbol{x}_{k'}) \sum_{k''=1}^{K} w_T(\boldsymbol{x}_j, \boldsymbol{x}_{k''})} \geq \eta\}. \quad (7)$$

Here η is a pre-defined constant.

After calculating $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d\}}, \boldsymbol{x}_j^{T\cup\{d\}}]$ and $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}]$, we can then derive $\Delta_d(x_i, x_j, T, \zeta)$. Finally, taking all possible subset $T \subset [D] \setminus \{d\}$ into account, the contribution (i.e., importance score) of the d-th feature to the prediction of ζ on $(\boldsymbol{x_i}, \boldsymbol{x_j})$ is given as

$$\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = \sum_{T\subset[D]\setminus\{d\}} \frac{|T|!(D-|T|-1)!}{D!} \Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta). \quad (8)$$

This equation captures the average marginal contribution of the d-th feature by averaging $\Delta_d(\boldsymbol{x}_i, \boldsymbol{x}_j, T, \zeta)$ over all the possible subset T. The value of $\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)$ reflects the importance of the d-th feature to the final pairwise prediction. Based on this fact, we can identify which features in $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ make key contributions to the final predicted result.

An alternative way to calculate the contribution. Besides Eq.(8), we also have another way to calculate the contribution of the d-th feature to the prediction of ζ on $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Let $\pi(D)$ be the set of all possible ordered permutations of the feature indices $\{1, 2, ..., D\}$. Let $\mathcal{O}$ be any permutation of the feature index $\{1, 2, ..., D\}$. For the permutation $\mathcal{O} \in \pi(D)$, we denote the set of features that precede d in $\mathcal{O}$ as $P_{\mathcal{O}}^d$. From Eq. (8), we know that $\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)$ is the average marginal contribution of d to any coalition of D assuming that all orderings are equal. Another way to calculate $\phi_i(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)$ is averaging the marginal contributions of the d-th feature to the set of its predecessors, where the average value is taken over all permutations equally. Thus, we have

$$\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = \frac{1}{D!} \sum_{\mathcal{O}\in\pi(D)} (\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{P_{\mathcal{O}}^d\cup\{d\}}, \boldsymbol{x}_j^{P_{\mathcal{O}}^d\cup\{d\}}] - \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{P_{\mathcal{O}}^d}, \boldsymbol{x}_j^{P_{\mathcal{O}}^d}]). \quad (9)$$

Next, we provide the theoretical analysis to show that the proposed ASIPair is the unique pairwise interpretation method with the desired theoretical properties, which strongly motivates the use of ASIPair for reliable pairwise interpretations.

Theorem 1. *The proposed ASIPair is the unique solution that satisfies the following properties:*

(1) Efficiency. The sum of the marginal contributions of all input features is equal to the pairwise function value, i.e. $\sum_{d=1}^{D} \phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = \zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

(2) Fairness. For all $T \subset \{1, 2, \cdots, D\} \setminus \{d_1, d_2\}$, *if* $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d_1\}}, \boldsymbol{x}_j^{T\cup\{d_1\}}] = \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d_2\}}, \boldsymbol{x}_j^{T\cup\{d_2\}}]$, *then we have* $\phi_{d_1}(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = \phi_{d_2}(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)$.

(3) Dummy. For all $T \subset \{1, 2, \cdots, D\} \setminus \{d\}$, *if* $\mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T\cup\{d\}}, \boldsymbol{x}_j^{T\cup\{d\}}] = \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{T}, \boldsymbol{x}_j^{T}]$, *then we have* $\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = 0$.

(4) Additivity. For any two pairwise decision functions ζ_1 *and* ζ_2, *we have that for each* $d \in [D]$, $\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta_1+\zeta_2) = \phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta_1) + \phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta_2)$, *where* $(\zeta_1 + \zeta_2)$ *is defined by* $(\zeta_1 + \zeta_2)(\boldsymbol{x}_i, \boldsymbol{x}_j) = \zeta_1(\boldsymbol{x}_i, \boldsymbol{x}_j) + \zeta_2(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

Note that all the four properties are reasonable in the context of pairwise learning. The efficiency property states that the total pairwise value $\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is divided among all of the features. This property makes it easier to compare features' contributions. The fairness property means that if two features always add the same marginal value to any subset to which they are added, they will be assigned equal contributions on the total pairwise value $\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)$. The dummy property states that if a feature never adds any marginal value, the contribution value of this feature will be assigned with zero. The additivity property shows that the solution to the sum of two pairwise models (ζ_1 and ζ_2) must be the sum of what it assigns to each of the two pairwise models. Although the proposed ASIPair method can effectively score the importance of the input features on the predicted result and provide theoretical guarantee, the Shapley value approach makes it computationally challenging. The computation complexity of ASIPair (in terms of the pairwise model evaluations) on all features' contributions (i.e., $\{\phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)\}_{d=1}^{D}$) is $O(D * 2^{D \log_2 D})$. To address this problem, we further propose a more efficient interpretation method for pairwise models in the next section.

Robust Approximation Interpretation Method for Pairwise Models

In this section, we propose a robust approximation interpretation method for pairwise models (RAIPair), which is motivated by that not all features are important and only a subset of features contain the discriminative information for the final predicted result. RAIPair is not only much more efficient than ASIPair but also robust to data noise.

Let $\boldsymbol{\phi} = (\phi_1(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta), \cdots, \phi_D(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta)) \in \mathbb{R}^D$ be the feature importance score vector for the testing instance pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Based on Theorem 1, we know that $\sum_{d=1}^{D} \phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) = \zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Then, we can calculate the average feature importance score over all the features as $\bar{\phi} = \zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)/D$. Suppose $\tilde{\boldsymbol{\phi}}^* \in \mathbb{R}^D$ denotes the deviation vector, in which the d-th element $\tilde{\phi}_d^*$ is calculated as $\tilde{\phi}_d^* = \phi_d(\boldsymbol{x}_i, \boldsymbol{x}_j, \zeta) - \bar{\phi}$. Obviously, we can get $\boldsymbol{\phi} = \bar{\phi}\boldsymbol{I}_D + \tilde{\boldsymbol{\phi}}^*$, where $\boldsymbol{I}_D$ is a D dimensional vector with all entries being one. For the non-discriminative features, their importance scores are close to the average value $\bar{\phi}$, and the deviation values of them are close to zero. In contrast, for the discriminative features, their importance scores deviate from the average value $\bar{\phi}$ far away.

Since the average value $\bar{\phi}$ can be easily calculated, to reduce the computational complexity, our goal here is to first optimize the deviation vector $\tilde{\boldsymbol{\phi}}^*$, and then derive the feature importance score vector based on $\boldsymbol{\phi} = \bar{\phi}\boldsymbol{I}_D + \tilde{\boldsymbol{\phi}}^*$. Considering that only a subset of features contain the discriminative information, we assume that $\tilde{\boldsymbol{\phi}}^*$ is s-sparse, and $\tilde{\boldsymbol{\phi}}^*$ is called s-sparse if $\|\tilde{\boldsymbol{\phi}}^*\|_{\mathcal{L}_0} \leq s$, where $\|\tilde{\boldsymbol{\phi}}^*\|_{\mathcal{L}_0} = \lim_{p\to 0} \sum_{d=1}^{D} |\tilde{\phi}_d^*|^p = \sum_{d=1}^{D} \mathbb{I}(\tilde{\phi}_d^* \neq 0)$. Specifically, the quantity $\|\tilde{\boldsymbol{\phi}}^*\|_{\mathcal{L}_0}$ computes the number of nonzero elements in the feature importance score vector $\tilde{\boldsymbol{\phi}}^*$. To calculate the sparse vector $\tilde{\boldsymbol{\phi}}^*$, we propose to solve the following optimization problem

$$\tilde{\boldsymbol{\phi}}^* = \tag{10}$$
$$\text{argmin}_{\tilde{\boldsymbol{\phi}}\in\mathbb{R}^D} \{\|\tilde{\boldsymbol{\phi}}\|_{\mathcal{L}_0} : \text{ s.t. } \|\boldsymbol{b} - \boldsymbol{A}(\bar{\phi}\boldsymbol{I}_D + \tilde{\boldsymbol{\phi}})\|_{\mathcal{L}_2} = 0\},$$

where $\boldsymbol{b} \in \mathbb{R}^M$ is an observed measurement vector, and $\boldsymbol{A} \in \mathbb{R}^{M\times D}$ is a random Bernoulli measurement matrix for which the number of rows is far less than that of columns (i.e., $M \ll D$). The entries of $\boldsymbol{A}$ take the value $\frac{1}{\sqrt{M}}$ or $-\frac{1}{\sqrt{M}}$ with equal probability. In Eq.(10), we aim to use much fewer measurements to calculate $\tilde{\boldsymbol{\phi}}^*$ and further derive the feature importance score vector $\boldsymbol{\phi}$.

However, the above $\mathcal{L}_0$-norm minimization formulation is NP-hard because it involves enumerative search and is computationally intractable for practical applications. Besides scalability, another important requirement for real-world applications is the robustness to noise, namely, the observation vector $\boldsymbol{b}$ may be corrupted by data noise. Without loss of generality, we assume that the measurement vector $\boldsymbol{b}$ is corrupted by noise of magnitude up to ϵ. To address the computationally intractable problem, we propose to use the convex relaxation by replacing the $\mathcal{L}_0$-norm with the $\mathcal{L}_1$-norm. To take the noise into account, we propose to relax the equality constraint as $\|\boldsymbol{b} - \boldsymbol{A}(\bar{\phi}\boldsymbol{I}_D + \tilde{\boldsymbol{\phi}})\|_{\mathcal{L}_2} \leq \epsilon$. Then, we can derive the following optimization problem

$$\hat{\boldsymbol{\phi}}^* = \tag{11}$$
$$\text{argmin}_{\hat{\boldsymbol{\phi}}\in\mathbb{R}^D} \{\|\hat{\boldsymbol{\phi}}\|_{\mathcal{L}_1} : \text{ s.t. } \|\boldsymbol{b} - \boldsymbol{A}(\bar{\phi}\boldsymbol{I}_D + \hat{\boldsymbol{\phi}})\|_{\mathcal{L}_2} \leq \epsilon\},$$

where $\epsilon > 0$ is a pre-defined noise level. Finally, $\tilde{\boldsymbol{\phi}}^* \in \mathbb{R}^D$ can be well approximated by $\hat{\boldsymbol{\phi}}^*$ based on Eq. (11). Note that the above problem is an underdetermined linear problem since $M \ll D$, and the $\mathcal{L}_1$-norm minimization solution is also the sparsest possible solution (Bruckstein, Donoho, and Elad 2009; Candes and others 2006). The problem can be recast as a linear program and can be solved by conventional methods such as interior-point methods. However, these methods suffer from poor scalability for real-world problems with large-scale data. To address this challenge, we propose to use the fast iterative shrinkage-threshold method (Beck and Teboulle 2009) to solve the above optimization problem, and the proposed RAIPair is summarized

in Algorithm 1. In this algorithm, we first estimate the measurement vector $\boldsymbol{b}$ from a set of random permutations (i.e., $\{\mathcal{O}_h\}_{h=1}^H$) (Step 1-12), and then derive the approximated importance score vector $\tilde{\phi}$ by solving the $\mathcal{L}_1$ minimization problem (Step 13-14). In Theorem 2, we also present the approximation error bound for the proposed RAIPair. The computational complexity of RAIPair on all features' contributions is $H * D = O(\log(\log D) * D)$, where D is the feature dimension. Thus, the computational complexity of RAIPair is much lower than that of ASIPair

Algorithm 1 The robust approximation interpretation method for pairwise models

Input: Pairwise model ζ, the number of measurements M, the test pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$, the number of permutations H, and the random Bernoulli matrix $\boldsymbol{A}$.

1: **for** h $\leftarrow 1$ to H **do**
2: Randomly select the permutation $\mathcal{O}_h \in \pi(D)$;
3: **for** $d \leftarrow 1$ to D **do**
4: $\Delta_d^h(\boldsymbol{x}_i, \boldsymbol{x}_j, P^d_{\mathcal{O}_h}, \zeta) = \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|x_i^{P^d_{\mathcal{O}_h}\cup\{d\}}, \boldsymbol{x}_j^{P^d_{\mathcal{O}_h}\cup\{d\}})] - \mathbb{E}[\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)|\boldsymbol{x}_i^{P^d_{\mathcal{O}_h}}, \boldsymbol{x}_j^{P^d_{\mathcal{O}_h}})]$;
5: **end for**
6: **for** $m \leftarrow 1$ to M **do**
7: $Y_{m,h} \leftarrow \sum_{d=1}^D \boldsymbol{A}_{m,d}\Delta_d^h(\boldsymbol{x}_i, \boldsymbol{x}_j, P^d_{\mathcal{O}_h}, \zeta)$;
8: **end for**
9: **end for**
10: **for** $m \leftarrow 1$ to M **do**
11: $b_m = \frac{1}{H}\sum_{h=1}^H Y_{m,h}$; // b_m is the m-th element in $\boldsymbol{b}$
12: **end for**
13: $\hat{\boldsymbol{\phi}}^* = \text{argmin}_{\hat{\boldsymbol{\phi}}\in\mathbb{R}^D}\{\|\hat{\boldsymbol{\phi}}\|_{\mathcal{L}_1} : \text{ s.t. } \|\boldsymbol{b} - \boldsymbol{A}(\bar{\phi}\boldsymbol{I}_D + \hat{\boldsymbol{\phi}})\|_{\mathcal{L}_2} \leq \epsilon\}$;
14: **return** the approximated feature importance score vector $\tilde{\boldsymbol{\phi}} = \bar{\boldsymbol{\phi}}\boldsymbol{I}_D + \hat{\boldsymbol{\phi}}^*$.

Theorem 2. *Assume that the range of the predictions made by the pairwise model $\zeta(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is $[-r, +r]$, and the restricted isometry constant δ_{2s} of the matrix $\boldsymbol{A} \in \mathbb{R}^{M\times D}$ satisfies $\delta_{2s} < \frac{3}{4+\sqrt{6}} \approx 0.465$. Let $\sigma_s(\boldsymbol{\phi})_{\mathcal{L}_1} := \inf\{\|\boldsymbol{\phi} - \Psi\|_{\mathcal{L}_1}, \Psi$ is s-sparse$\}$, $\epsilon > 0$, $0 < \delta < 1$, and C be a universal constant. Then, if $M \geq C(0.465)^{-2}(2s\log(D/(2s)) + \log(2/\delta))$ and $\frac{2r^2}{\epsilon^2}\log\frac{4M}{\delta} \leq H$, we then can derive*

$$\|\tilde{\boldsymbol{\phi}} - \boldsymbol{\phi}\|_{\mathcal{L}_2} = \|\hat{\boldsymbol{\phi}}^* - \hat{\boldsymbol{\phi}}\|_{\mathcal{L}_2} \leq \Phi_1\epsilon + \Phi_2\frac{\sigma_s(\boldsymbol{\phi})_{\mathcal{L}_1}}{\sqrt{s}}, \quad (12)$$

where ϵ denotes the noise amount, $\Phi_1 \in \mathbb{R}$ and $\Phi_2 \in \mathbb{R}$ are two constants that only depend on δ_{2s}. Note that H denotes the number of random permutations used to estimate the measurement vector $\boldsymbol{b}$.

Based on Theorem 2, we can bound the error between the proposed ASIPair and its approximated version (i.e., RAIPair). In fact, both ASIPair and RAIPair use the same strategy to calculate the marginal function Δ_d, which can be seen in Eq.(2) and Step 4 of Algorithm 1. The difference between ASIPair and RAIPair is that RAIPair uses very few operations to average Δ_d to approximate ASIPair.

Experiments

We conduct experiments on both real-world and synthetic datasets to evaluate the performance of the proposed interpretation methods. All the experiments are conducted 10 times and we report the average results.

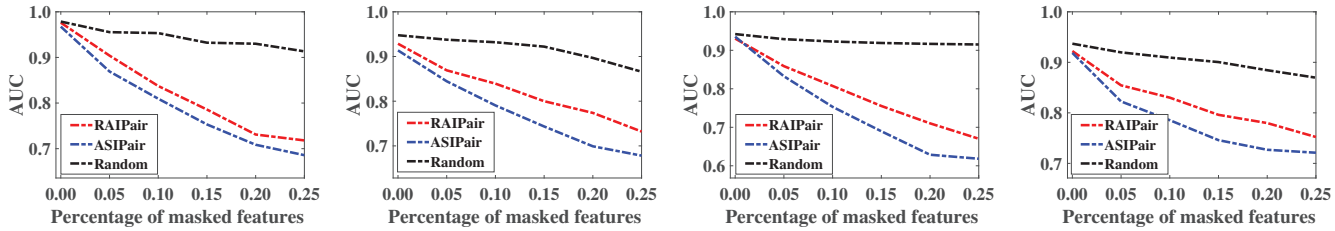
Datasets For real-world datasets, we adopt four UCI datasets (i.e., Heart, Diabetes, Parkinson and Ionosphere), and the MNIST 1V9 dataset (LeCun et al. 1998) that is a subset of the 784-dimensional MNIST set. The statistical information of these real-word datasets is described in Table 1. For the synthetic dataset, we use the following method to generate the data: We first generate N instances $\{\boldsymbol{x}_i\}_{i=1}^N$, where $\boldsymbol{x}_i$ is a D-dimensional feature vector in which each element is randomly generated in range $(-1, 1)$. Then we build a linear classifier with the weight vector $\boldsymbol{w}$ in which each element $w_i \sim U(-0.5, 0.5)$. Finally, we use the linear classifier to generate the label of each instance. For each dataset, we randomly select 80% of the instances as the training set to train the pairwise model, and take the rest instances as the test set.

Table 1: The statistics of the datasets.

Dataset	Size	Dimension
Heart	303	23
Diabetes	768	9
Parkinson	195	22
Ionosphere	351	34
MNIST	2,134	784

Performance Measure We evaluate the performance of the proposed interpretation methods through observing the change of the predicted results after masking a proportion of the top features ranked by the learned feature importance scores. Specifically, given a trained pairwise model and a test instance pair, both the proposed ASIPair and RAIPair can generate a vector of importance scores that reflects all features' contributions to the pairwise prediction on this test pair. When evaluating the performance of each proposed method, we first rank the importance scores and mask a proportion of the top ranked features. Then, we measure the change of the result predicted by the pairwise model before and after masking the features. The larger the predicted result changes, the more important the masked features are. In addition, considering that there is no existing interpretation method designed for pairwise learning, we adopt the random masking method as the baseline, in which we randomly select a proportion of features and then mask them.

Interpretation for AUC Maximization We first study the performance of the proposed interpretation methods on a widely used AUC maximization model, i.e., OPAUC (Gao et al. 2013), which aims to maximize the AUC metric by going through the training data only once without storing the entire training dataset. Here we evaluate the performance of



(a) Heart (b) Diabetes (c) Parkinson (d) Inosphere

Figure 1: The AUC metric w.r.t the percentage of masked features.

the proposed methods through observing the change of AUC metric before and after masking the top features ranked by the calculated importance scores. The AUC metric of a pairwise model is equal to the probability that the model ranks a randomly chosen positive instance higher than a randomly chosen negative instance (Khalid, Ray, and Chitsaz 2016). The lower the AUC metric, the larger the change of the predicted results, the better the proposed interpretation method. In this experiment, we vary the percentage of masked features over the total number of features from 0.001 to 0.25, and the AUC metric on the four UCI datasets are shown in Figure 1. As we can see, compared with masking a proportion of randomly selected features, masking the top ranked features derived based on our proposed interpretation methods have more effect on the predicted results, which means both ASIPair and RAIPair can effectively identify important features that make key contributions to the predicted results. Additionally, this figure also shows that the performance of ASIPair is little better than that of RAIPair.

Efficiency We also evaluate the efficiency of the proposed interpretation methods. In this experiment, we adopt a widely used metric learning model, i.e., LowRank (Zhan et al. 2016), which aims to learn a metric that can measure the similarity degree between a pair of instances. In this experiment, we generate several synthetic datasets by varying the value of D from 2 to 9. The size of each synthetic dataset (i.e., N) is set as 1000. We then evaluate the running time of ASIPair and RAIPair on each dataset, and the average result on all testing instance pairs is shown in Figure 2. From this figure, we can see that the running time of RAIPair is polynomial with respect to the input feature dimension D while that of ASIPair is approximately exponential with respect to D. When the number of features increases, RAIPair shows great advantage in running time, which verifies our conclusion that RAIPair is much more efficient than ASIPair.

Sparsity In addition, we evaluate the performance of the proposed RAIPair on sparse feature selection using the MNIST dataset. The pairwise model to be interpreted is OPAUC. We randomly select four testing instance pairs and report the deviation score of each feature (i.e., $\hat{\phi}_d^* \in \hat{\boldsymbol{\phi}}^*$) calculated by RAIPair in Figure 3. The results in this figure show that RAIPair can effectively select a subset of features that make key contributions to the predicted result. Take Figure 3a as an example. We can see the deviation scores of most features are close to 0 and only a subset of features

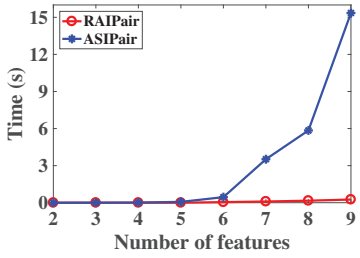


Figure 2: Running time of ASIPair and RAIPair.

have large deviation scores, which means the importance scores of most features are around the average value (i.e., $\bar{\phi}$) and only a subset of features play an important role in providing discriminative information for the final predicted result. Based on Figure 3a, we can also know only a subset of features contain discriminative information.

Visualization Last but not least, we provide some visualization results to further evaluate the effectiveness of the proposed methods. In this experiment, we use RAIPair to interpret the predictions made by the AUC maximization model OPAUC on the MNIST 1V9 dataset. The results for one testing instance pair (a picture for digital 9 and a picture for digital 1) are shown in Figure 4. Figure 4a shows the correctly classified testing instance pair, which means that the positive instance (the digit 9) ranks higher than the negative instance (the digit 1). After applying RAIPair to this pair, we can get the importance score vector associated with the underlying features of this pair. We then rank these importance scores and select the 8% of the top-ranked features, which are highlighted with red color in Figure 4b. In Figure 4c, we use blue rectangles to highlight the salient parts of the selected features that make key contributions to the pairwise prediction. As we can see, the proposed RAIPair can accurately capture the salient parts of the input features for the pairwise predicted result, and these parts agree well with the empirical intuition of humans.

Related Work

Although pairwise learning has been well studied in many works (Ying, Wen, and Lyu 2016; Natole, Ying, and Lyu 2018), there is no existing work that considers how to interpret the predictions made by the learned pairwise models. Recently, a wide variety of interpretation methods have been

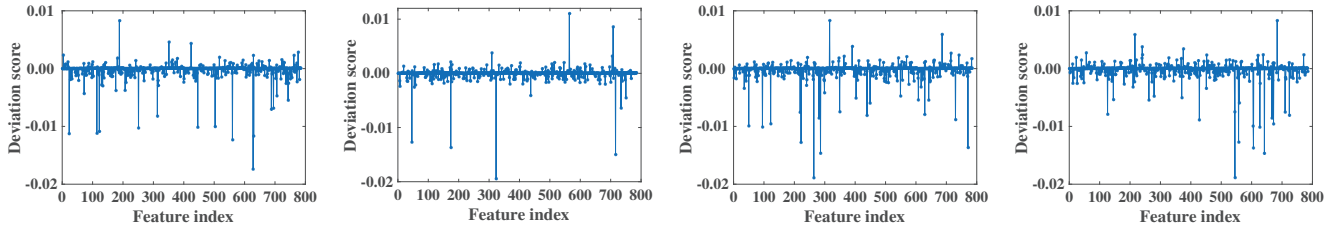


Figure 3: The calculated deviation scores for testing instance pairs on MNIST dataset. The results in (a)-(d) are for four different instance pairs.

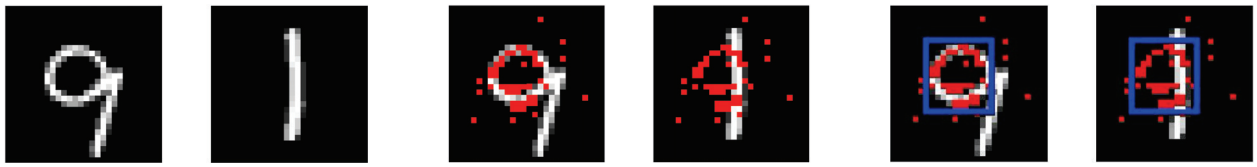


Figure 4: Visualization results generated by the proposed RAIPair on the MNIST dataset.

developed to provide explanations for the pointwise learning models (e.g., classification model) through scoring the importance of each input feature for a given instance (Ribeiro, Singh, and Guestrin 2016; Datta, Sen, and Zick 2016; Shrikumar, Greenside, and Kundaje 2017; Štrumbelj and Kononenko 2014; Lundberg and Lee 2017; Chen et al. 2018; Ancona, Öztireli, and Gross 2019; Kononenko and others 2010; Koh and Liang 2017; Grün et al. 2016). Among these pointwise interpretation methods, the Shapley-value-based methods have drawn significant attention as they can provide strong theoretical foundations and better agree with the human intuition (Štrumbelj and Kononenko 2014; Lundberg and Lee 2017; Chen et al. 2018; Ancona, Öztireli, and Gross 2019; Kononenko and others 2010; Selvaraju et al. 2017; Zeiler and Fergus 2014). However, these pointwise interpretation methods do not take into account pairwise input and cannot be directly used in pairwise learning. To calculate the importance scores of the input features, these methods usually need a pre-defined reference vector and require all the instances to use the same reference vector, which is unreasonable for pairwise models. Furthermore, these methods usually assume that the input features do not correlate with each other. However, in practice, the features of an instance may correlate with each other and the correlations can also affect the predicted results. Thus, we take into account feature correlations when we design the interpretation methods for pairwise models.

Considering that Shapley value approach is usually computationally challenging, existing works also take measures to reduce the computational cost. Specifically, the authors in (Kononenko and others 2010; Lundberg and Lee 2017; Štrumbelj and Kononenko 2014) propose various sampling-based approximation methods. However, these sampling-based methods may suffer from high variance when the number of instances to be collected is limited. The authors in (Chen et al. 2018) develop two approximation methods based on the assumption that the input features have an underlying graph structure. By assuming that the hidden layer is distributed with an isotropic Gaussian, the authors in (Ancona, Öztireli, and Gross 2019) propose an approximation method for deep neural networks. Different from these methods, our proposed approximation method RAIPair is more general and does not make any assumptions on the input data structure.

In addition, this work is significantly different from existing pairwise feature selection methods (Gao et al. 2014; Ying, Huang, and Campbell 2009) for pairwise learning. Existing pairwise feature selection methods aim to alter the training phase to learn a subset of features that are relevant to the targeted model. Also, even for these selected features, they cannot distinguish their relative relevance scores. However, our proposed interpretation methods only involve the testing stage, and aim to interpret each individual pairwise prediction that is made by the trained pairwise model.

Conclusions

In this paper, we investigate how to enable interpretation in pairwise learning. Specifically, we first propose a novel adaptive interpretation method for pairwise learning (i.e., ASIPair), based on which a vector of importance scores associated with the underlying features of a testing instance pair can be adaptively calculated, and these scores can be used to indicate which features make key contributions to the final prediction. Considering that the proposed ASIPair is computationally challenging, we further propose a novel robust approximation interpretation method for pairwise models (i.e., RAIPair). This method is not only much more efficient but also robust to data noise. Theoretical analysis and extensive experiments demonstrate the effectiveness of the proposed interpretation methods for pairwise learning.

Acknowledgments

This work is supported in part by the US National Science Foundation under grants IIS-1924928, IIS-1938167 and OAC-1934600. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Ancona, M.; Öztireli, C.; and Gross, M. 2019. Explaining deep neural networks with a polynomial time algorithm for shapley values approximation. *arXiv preprint arXiv:1903.10992*.

Beck, A., and Teboulle, M. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* 2(1):183–202.

Boissier, M.; Lyu, S.; Ying, Y.; and Zhou, D.-X. 2016. Fast convergence of online pairwise learning algorithms. In *Artificial Intelligence and Statistics*, 204–212.

Bruckstein, A. M.; Donoho, D. L.; and Elad, M. 2009. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM review* 51(1):34–81.

Candes, E. J., et al. 2006. Compressive sampling. In *Proc. of the international congress of mathematicians*.

Chen, J.; Song, L.; Wainwright, M. J.; and Jordan, M. I. 2018. L-shapley and c-shapley: Efficient model interpretation for structured data.

Datta, A.; Sen, S.; and Zick, Y. 2016. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, 598–617. IEEE.

Gao, W.; Jin, R.; Zhu, S.; and Zhou, Z.-H. 2013. One-pass auc optimization. In *ICML*, 906–914.

Gao, X.; Hoi, S. C.; Zhang, Y.; Wan, J.; and Li, J. 2014. Soml: Sparse online metric learning with application to image retrieval. In *Twenty-eighth AAAI conference on artificial intelligence*.

Grün, F.; Rupprecht, C.; Navab, N.; and Tombari, F. 2016. A taxonomy and library for visualizing learned features in convolutional neural networks. *arXiv preprint arXiv:1606.07757*.

Huai, M.; Miao, C.; Li, Y.; Suo, Q.; Su, L.; and Zhang, A. 2018a. Metric learning from probabilistic labels. In *Proc. of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1541–1550.

Huai, M.; Miao, C.; Suo, Q.; Li, Y.; Gao, J.; and Zhang, A. 2018b. Uncorrelated patient similarity learning. In *Proc. of the 2018 SIAM International Conference on Data Mining*, 270–278. SIAM.

Huai, M.; Xue, H.; Miao, C.; Yao, L.; Su, L.; Chen, C.; and Zhang, A. 2019. Deep metric learning: the generalization analysis and an adaptive algorithm. In *Proc. of the 28th International Joint Conference on Artificial Intelligence*.

Khalid, M.; Ray, I.; and Chitsaz, H. 2016. Confidence-weighted bipartite ranking. In *International Conference on Advanced Data Mining and Applications*, 35–49. Springer.

Koh, P. W., and Liang, P. 2017. Understanding black-box predictions via influence functions. In *ICML*.

Kononenko, I., et al. 2010. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research* 11(Jan):1–18.

LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.; et al. 1998. Gradient-based learning applied to document recognition. *Proc. of the IEEE* 86(11):2278–2324.

Lundberg, S. M., and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.

Natole, M.; Ying, Y.; and Lyu, S. 2018. Stochastic proximal algorithms for auc maximization. In *ICML*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proc. of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proc. of the IEEE International Conference on Computer Vision*, 618–626.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *ICML*, 3145–3153. JMLR. org.

Štrumbelj, E., and Kononenko, I. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems* 41(3):647–665.

Suo, Q.; Zhong, W.; Ma, F.; Ye, Y.; Huai, M.; and Zhang, A. 2018. Multi-task sparse metric learning for monitoring patient similarity progression. In *ICDM*, 477–486.

Tang, J., and Wang, K. 2018. Ranking distillation: Learning compact ranking models with high performance for recommender system. In *Proc. of the 24th SIGKDD International Conference on Knowledge Discovery & Data Mining*.

Xie, P.; Zhang, H.; Zhu, Y.; and Xing, E. P. 2018. Nonoverlap-promoting variable selection. In *ICML*.

Yao, L.; Li, S.; Li, Y.; Huai, M.; Gao, J.; and Zhang, A. 2018. Representation learning for treatment effect estimation from observational data. In *Advances in Neural Information Processing Systems*, 2633–2643.

Ying, Y.; Huang, K.; and Campbell, C. 2009. Sparse metric learning via smooth optimization. In *Advances in neural information processing systems*, 2214–2222.

Ying, Y.; Wen, L.; and Lyu, S. 2016. Stochastic online auc maximization. In *Advances in neural information processing systems*, 451–459.

Zeiler, M. D., and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*, 818–833. Springer.

Zhan, M.; Cao, S.; Qian, B.; Chang, S.; and Wei, J. 2016. Low-rank sparse feature selection for patient similarity learning. In *ICDM*, 1335–1340. IEEE.