Human Understandable Explanation Extraction for Black-box Classification Models Based on Matrix Factorization

Jaedeok Kim and Jingoo Seo
Artificial Intelligence Team, Software R&D Center, Samsung Electronics
56 Seongchon-gil, Seocho-gu, Seoul, Republic of Korea

Abstract

In recent years, a number of artificial intelligent services have been developed such as defect detection system or diagnosis system for customer services. Unfortunately, the core in these services is a black-box in which human cannot understand the underlying decision making logic, even though the inspection of the logic is crucial before launching a commercial service. Our goal in this paper is to propose an analytic method of a model explanation that is applicable to general classification models. To this end, we introduce the concept of a contribution matrix and an explanation embedding in a constraint space by using a matrix factorization. We extract a rule-like model explanation from the contribution matrix with the help of the nonnegative matrix factorization. To validate our method, the experiment results provide with open datasets as well as an industry dataset of a LTE network diagnosis and the results show our method extracts reasonable explanations.

INTRODUCTION

In recent years, a number of artificial intelligent services have been developed such as defect detection system or diagnosis system for customer services. The core in these services consists of a classification model learned from historical data that classified by domain experts. To have a better model many advanced techniques have been applied like ensemble, boosting, or deep learning, and these techniques have helped in improving the performance of a classification model (Kotsiantis, Zaharakis, and Pintelas 2007). Unfortunately, such model has a black-box problem and human cannot understand how the model makes a decision. The black-box model has difficulty to give a trust for users even if the model shows a low classification error.

This is a problem of a model inspection. From the viewpoint of service providers, the inspection of the model behavior is crucial before launching their commercial services. Since service providers have potentially risks in applying a black-box model, they spend much time to verify a model until its reliability is proved and it increases development costs significantly. So the service providers have still preferred a hybrid of a machine learning based model and a rule based model in their services.

We consider the model inspection problem of a classification model. To solve this problem, we propose a method to extract rule-like explanations of a classification model to account for the underlying decision making constraint. Our approach can help a service provider understand the decision making logic of the model and quickly decide whether the model is acceptable or not. As a consequence, our result will help significantly reduce the efforts in the model inspection.

Explaining a black box model is not an easy task in general because of the variety of algorithms to build a classification model. There have few works to analyze how a model predicts a result for a given instance. To understand the reason of a classification result in an image classification problem, visualization techniques such as the saliency map have been used (Simonyan, Vedaldi, and Zisserman 2014; Zintgraf et al. 2017). Similar to the silence map, a trust problem on a prediction result has been tackled (Ribeiro, Singh, and Guestrin 2016). Their approach can explain the reason of prediction results by a model in the instance level, but it is not enough to inspect the characteristics of a model.

Our method is based on their results. We will extract common decision constraints from explanations of classification results by using the nonnegative matrix factorization. When we have a set of explanations of a classification result, the explanations shares the same characteristics if the model has underlying decision making constraints. For instance, when experts classify system diagnosis issues into several categories, they have their own decision making constraints even if they cannot recognize it explicitly. The constraints should be reflected in the explanations of instances.

The main contribution in this paper is that we propose an analytic method of a model explanation that is applicable to general classification models. We introduce the concept of a contribution matrix and an explanation embedding in a constraint space by using the matrix factorization. We experimentally show that the embedded explanations are well clustered according to the classification category of instances. By using the explanation embedding, we extract a model explanation of the rule-like form. We also perform experiments with open datasets as well as an industrial dataset to show the validity of our approach in practice. In particular, many of industrial datasets, especially defect detection or diagnosis systems, consist of numerical attributes each of them has its own meaning. Our method therefore will focus on dataset having numeric attributes.
Related Works

Understanding a model is one of the key issues in the machine learning field. However, understanding a black-box model is difficult in general because of the variety of algorithms and the dependency of data characteristics. So the decision tree or the support vector machine have been still popular and their variations have been published over decades because those algorithms have intuitive structures to be interpreted (Witten et al. 2016). Unfortunately, they do not guarantee a sufficiently good performance in practice and we often need to apply advanced techniques to improve the performance, such as ensemble or boosting, which makes a model be a black box like a deep neural network that is the promising technique nowadays.

Visualization techniques such as the saliency map have been often used in order to understand classification results in the image classification problem. To visualize the evidences of an image classification model, (Simonyan, Vedaldi, and Zisserman 2014) measures the sensitivity of classifications and (Zintgraf et al. 2017) analyzes the activation difference in the marginal distribution of a small area. These approaches are specified to image classifiers, while they are not easy to be extended to the other applications.

To overcome the difficulty of model understanding, the explanation of a black box model has been studied in recent years. Lipton discussed the meaning of interpretability and the properties of interpretable models (Lipton 2016). Tuner introduced several axioms to define the concept of explanation such as the eligibility of explanation in terms of the conditional probability (Turner 2016). A model explanation system was also developed to explain results of a black box model based on the axioms. Ribeiro et al. tackled the trust problem and proposed a method to explain a classification result (Ribeiro, Singh, and Guestrin 2016). The key idea is that a classification model can be locally approximated by perturbing each instance and the approximation can explain how much each feature affects the decision. The local perturbation is also used in (Fong and Vedaldi 2017) for interpreting prediction results in the image classification domain. In (Shrikumar, Greenside, and Kundaje 2017), Shrikumar et al. tackled such issue and proposed a method to determine the important features by backpropagating the contributions of neurons. Ross et al. develop a method to train a model with the right reasons (Ross, Hughes, and Doshi-Velez 2017). For a differentiable classifier such as neural network, they considered the first order derivative of the classifier as an explanation and designed a loss function with a consideration of an explanation. In the above literatures, only instance level explanation has been focused and a model level explanation has been overlooked.

A model level explanation necessarily needs to understand the underlying decision making constraint of a given classification model. Kim et al. proposed a simplification method of Bayesian network model (Kim, Rudin, and Shah 2014). Rule extraction methods have been proposed in (Núñez, Angulo, and Catalá 2002), but they are restricted in the support vector machine. Bastani et al. try to understand the reasoning process of a black box model by extracting a decision tree that approximates the classification model (Bastani, Kim, and Bastani 2017).

Explanation Modeling

In this section, we would develop a mathematical model to describe precisely a model explanation.

A classification model is a function \( f : S \rightarrow C \) where \( S \) is a space of instances and \( C = \{1, \ldots, C\} \) is a category space. Since we want to focus on a dataset consisting of only numerical attributes without any categorical attribute, we suppose \( S \) is in \( \mathbb{R}^m \) where \( m \) is the number of attributes. Let \( D \) be a subset of \( S \) of size \( N \), called a dataset. Let \( R_c(D) \) be the set of instances in a dataset \( D \subset S \) that are classified by \( c \), i.e.,

\[
R_c(D) := \{ x \in D : f(x) = c \}.
\]

Also, we let \( R(D) := \{ R_1(D), \ldots, R_C(D) \} \) be a partition of \( D \). For convenience, we will use \( R_c \) instead of \( R_c(D) \) unless there is ambiguity.

The purpose of a model explanation is to understand classification constraints for each category. For numerical attributes the easiest way to intuitively understand a constraint is checking whether a value of an attribute is in a certain interval or not. So it is enough to find rough constraints to approximate each of the partition \( R \), which can be done as the following.

\[
R_c \approx \bigcup_i B_{c,i} := B_c
\]

where \( B_{c,i} \) is a subset of the form

\[
B_{c,i} := \{(x_1, \cdots, x_m) \in S : a_j < x_j \leq b_j, j = 1, \cdots, m\} = \prod_{j=1}^m [a_j, b_j],
\]

called a rectangle. Here, \( a_j \) or \( b_j \) can have the infinity values \( -\infty \) or \( \infty \) which induces a half bounded interval or an unbounded interval.

We then define a model explanation of \( f \) by a collection

\[
B := \{B_1, \cdots, B_C\}
\]

of the set \( B_c \) of rectangles. In addition, we call \( B_c \) an model explanation of the category \( c \).

There is an advantage of approximating each \( R_c \) instead of approximating the function \( f \). Each component \( B_{c,i} \) of \( B_c \) is a product of intervals. So the component \( B_{c,i} \) is directly matched with a rule-like explanation which is very simple to understand. In the existing literatures, a decision tree has been considered as an interpretable model and algorithms to extract a decision tree from a classification model have been developed (Bastani, Kim, and Bastani 2017). A decision tree may have an advantage of understanding reasoning process since in the decision tree a path from the root to a leaf intuitively represents a process of decision making. However, a decision tree is still hard to understand as a simple rule because of the instability of a tree structure (Li and Belford 2002).

Extracting an approximation model \( f^{(a)} \) is about finding a hard constraint on decision boundaries. In this approach,
every instance has one category, that is, $(f^{(a)})^{-1}(c) := \{ x \in S : f^{(a)}(x) = c \}$, $c \in \mathcal{C}$, are mutually disjoint. To do this, $f^{(a)}$ should decide which category is more suitable for an instance even if it lies on a boundary between the areas of two different categories. It follows that $f^{(a)}$ needs more complex constraints as the complexity of $f^{(a)}$ increases. Consequently, it leads barely understandable explanations, it is unnecessary to have a very accurate approximation of $f$.

**Performance Measure**

In order to measure the quality of a model explanation $B$, we use the $F_1$ score. Note that a model explanation $B_c$ of a category $c$ induces a binary classification $\hat{f}_c$ such that $\hat{f}_c(x) = c$ if $x \in B_c$. Consider a restriction $f_c$ of $f$ to a binary classification, i.e., $c$ and others. With considering a classification result by $f_c$ as a true value, we use a relative accuracy of $\hat{f}_c$ to $f_c$. Then, the performance of $B_c$ over a dataset $D$ is defined by the $F_1$ score of $\hat{f}_c$ relative to $f_c$ over the dataset $D$

$$F_1(B_c) := F_1(f_c, \hat{f}_c).$$

Finally, the performance of a model explanation is defined by

$$F_1(B) := \frac{1}{C} \sum_{c \in \mathcal{C}} F_1(B_c). \tag{2}$$

Note that there is no dependency between $F_1(B_c)$ and $F_1(B_{c'})$ for $c \neq c'$. The maximization of $F_1(B)$ is achievable by solving subproblems each of which maximizes the value of $F_1(B_c)$ for each category $c$. So we decompose our problem of finding $B$ into $c$ subproblems. Therefore, we tag a category $c$ as our target category and then assume $f$ is a binary classification model of two labels, say $c$ (the target category) and 0 (others), through the following sections.

**Model Explanation Extraction**

Extracting a model explanation consists of two main steps. Figure 1 summarizes the flow of our method. At first, an explanation for each instance will be generated by measuring how much an attribute of the instance contributes a classification result. The next step will construct a contribution matrix and then extract the common behaviors in the contribution matrix with the help of nonnegative matrix factorization. The detail of each step will be explained in the following subsections.

**Interpretable Instance Explanation**

To understand the behavior of a classification model, we first need to explain which attributes of an instance mainly affect a result and measure the amount of their impacts in a proper way. To this end, we consider a local perturbation of $f$ at a given instance $x$. In this paper, the perturbation is applied to interpretable features instead of attributes of an instance borrowing the idea introduced in Ribeiro, Singh, and Guestrin (2016). The advantage of this approach is that the complexity of explanations is controllable by restricting the number of interpretable features.

The problem of finding an instance explanation can be formulated as in the follows. We assume that $f$ is decomposed by an embedding $\phi$ to the interpretable space $\mathcal{F}$ and an interpretable model $\hat{f} : \mathcal{F} \to \mathcal{C}$ satisfying $f = \phi \circ \hat{f}$.

$$S \xrightarrow{\phi} \mathcal{F} \xrightarrow{\hat{f}} \mathcal{C}$$

Let $\mathcal{F}$ be an interpretable feature space of finite dimension whose dimension represents an interpretable feature. For numerical attributes, we may consider an interval of attribute value as an interpretable feature explained in the previous section. Since we are able to deal with only finite number of features, the domain of attributes should be discretized.

Recalling that a rectangle has the form of a product of intervals, we will use an indicator of the form $1_{[x_i \leq b]}$ as an interpretable feature. Only upper bounded interval is enough since $1_{[x_i \leq b]}(x) = 0$ implies $x_i > b$. Now we denote by $\mathcal{F}$ the set of indicators of a half-bounded interval. Then $\mathcal{F} = \{0,1\}^M$ is the interpretable space where $M := |\mathcal{F}|$. That is, for an element $z = (z_1, \ldots, z_M) = \phi(x)$ in $\mathcal{F}$, $z_i = 1$ implies $1_{[x_j \leq b]}(x) = 1$ for the corresponding indices $j_i$ and $b_i$.

Each feature will be directly used to classify categories without any further computation for the understandability of $\hat{f}$. So we restrict $\hat{f} : \mathcal{F} \to \mathcal{C}$ to a binary classification model whose decision boundary is given by a linear function $z \mapsto \phi^T z$ where $\phi = (\phi_0, \ldots, \phi_{M-1})^T$, that is, $z$ is classified by the target category $c$ if and only if $\hat{f}(z) > 0$. Then to find a suitable $\hat{f}$ for a given instance $x$ an optimization problem can be formulated as the following.

$$\hat{f}^* = \arg\min_\hat{f} J_x(f, \hat{f}) \tag{3}$$

Here $J_x$ is a distance function between two classification models $f$ and $\hat{f}$ over a perturbation set of $x$. If we let $\phi_0^T z$ be the decision boundary of $\hat{f}^*(z)$, its normal vector $\phi_0^T = (\phi_{0,1}, \ldots, \phi_{0,M-1})^T$ is said to be an explanation of the
instance $x$. We call $\phi_{i,j}^*$ the contribution of the $i$th feature on $x$ with respect to $f$. If $\phi_{i,j}^* > 0$, the $j$th feature $[x_{j,i} \leq b_i]$ positively affects the decision of $x$. Otherwise, $\phi_{i,j}^* < 0$ implies the $i$th feature negatively affects. In the later section, with the help of LIME (Ribeiro, Singh, and Guestrin 2016), we will evaluate $\phi_{i,j}^*$ by defining $J_f$ with the $F_1$ score that represents the relative difference between the results of $f$ and those of $f$.

Moreover, collecting all instance explanation we define the contribution matrix of a model $f$ with respect to a set $D$ by

$$\Phi_f(D) := \begin{bmatrix} \phi_{1,1} & \cdots & \phi_{1,N} \\ \vdots & \ddots & \vdots \\ \phi_{M,1} & \cdots & \phi_{M,N} \end{bmatrix}. \quad (4)$$

Here, $\phi_{f} = (\phi_{f,1,\ldots,\phi_{f,M}^T})^T$ is the instance explanation of $x_j \in D$, i.e., the solution of (3). For convenience an contribution matrix will be denoted by $\Phi_f$ instead of $\Phi_f(D)$ if there is no ambiguity.

**Model Explanation from Contribution Matrix**

The key intuition to extract a model explanation $B$ is the contribution matrix $\Phi_f$ contains the underlying constraints of decision making. By definition of $\Phi_f$, the $i$th row of the matrix $\Phi_f$ is the contributions of the $i$th feature. So the column vectors of $\Phi_f$ will be grouped into several clusters according to the decision making constraints of the classification model $f$, which can be found by using a matrix factorization. In particular, the nonnegative matrix factorization (NMF) can find base vectors whose elements are positive and represent how much a feature affects. As a consequence, rule-like explanations can be achievable by interpreting those base vectors.

The contribution matrix $\Phi_f$ contains both positive and negative values while we need a nonnegative matrix to apply the NMF. Let us consider a decomposition $\Phi_f = \Phi^+ - \Phi^-$ where $\Phi^+ = \phi_{i,j}^+$ is the positive matrix of $\Phi$ defined by $\phi_{i,j}^+ = \phi_{i,j} 1_{\phi_{i,j} > 0}$. Analogously, $\Phi^-$ is the negative matrix of $\Phi$. Instead $\Phi_f$, we consider a transformed contribution matrix $\bar{\Phi}_f$ of dimension $2M \times N$,

$$\bar{\Phi}_f := \begin{bmatrix} \Phi^+ \\ -\Phi^- \end{bmatrix},$$

whose elements are all nonnegative. $\phi_{i,j}^-$ is the negative contribution of the $i$th feature $[x_{j,i} > b_i]$ to the $j$th instance, which means the $x_{j,i}$ is more preferable to be in $[x_{j,i} > b_i]$ rather than in $[x_{j,i} \leq b_i]$. We take the complementary feature $z_i = 1 - 1_{[x_{j,i} \leq b_i]}$ of $z_i$ as the corresponding feature of the $i$th row of $\Phi^-$. We thus have 2M features.

We now consider the NMF of the nonnegative matrix $\bar{\Phi}_f$. Let $k$ be an integer such that $k \leq 2M, N$ and $W$ and $H$ be nonnegative matrices of dimension $2M \times k$ and $k \times N$, respectively, such that $W$ and $H$ are the solution of

$$\min_{W,H} \| \bar{\Phi}_f - WH \|.$$  

We then have

$$\phi_{j} = Wh_j = \sum_{l=1}^{k} h_{l,j} w_l \quad (5)$$

where $h_j = (h_{1,j}, \cdots, h_{k,j})^T$ is the $j$th column vector of $H$ and $w_l = (w_{1,l}, \cdots, w_{2M,l})^T$ is the $l$th column vector of $W$.

A vector $w_l$ can be interpreted as underlying decision making constraint of the classification model $f$. The value of an element $\phi_{i,j}^*$ in $\Phi_f$ represents the relationship between the $i$th feature and the classification result of the $j$th instance. Each $w_l$ is a base vector of the contribution matrix $\Phi_f$. So $w_l$ contains information of the feature combination that simultaneously affects the decision making of a category. Hence, we call the column space of $W$ a constraint space and $w_l$ a base vector of the constraint space.

Consider a base vector $w_l$. Let $\theta_w$ be a threshold that filters out meaningless features from the base vector $w_l$. We then have constraints by taking every feature that satisfies $w_{l,i} > \theta_w$. For instance, if $w_{l,i} > \theta_w$ and $1 \leq i \leq M$, we will take the corresponding constraint $[x_{j,i} \leq b_i]$, and if $M < i \leq 2M$, $[x_{j,i-M} > b_i-M]$ will be taken. Combining all constraints we obtain a rectangle of a underlying decision making constraint

$$B(w_l, \theta_w) := \prod_{i=1}^{M} [x_{j,i} \leq b_i]^{1[w_{l,i} > \theta_w]} [x_{j,i} > b_i]^{1[w_{l,i+M,i} > \theta_w]}.$$  

For convenience $A^0, A \subset S$, denotes the entire space $S$ of instances. If the value of $\theta_w$ becomes smaller, $B(w_l, \theta_w)$ takes the more constraints and the rule will be more complex. On the other hand, a larger value $\theta_w$ yields a less complex constraint $B(w_l, \theta_w)$.

A column vector $h_j = (h_{1,j}, \cdots, h_{k,j})^T$ of $H$ can be geometrically interpreted as a projection of $\phi_{j}$ onto the constraint space. Specifically, $h_{i,j}$ is the effect of the $j$th base vector on the $j$th instance. We call the column vector $h_j$ an embedded explanation. If there is a common decision making constraint in classification, the same rule should be applied to determine a category. Hence, the embedded explanations are expected to be concentrated in several clusters and almost all vectors in each cluster are labeled by the same category. In the later section, we will validate this assumption with experimental results.

Suppose there are $r$ clusters of embedded explanations and denote by $G_1, \cdots, G_r$. Since we have focused on a binary classification, the most frequent label in a cluster is $c$ (the target category) or $0$ (the others). Let $G_1, \cdots, G_r$ be the clusters in which elements are mostly labeled by the target category $c$. Let a vector $g_i = (g_{i,1}, \cdots, g_{i,k})$ be a centroid of $G_i$. Then the vector $g_i$ is the representative of $G_i$. This can be interpreted as follows. In $G_i$, $i \leq r$, the $l$th base vector $w_l$ contributes to the category $c$ with weight $g_{i,l}$ in average. Suppose an instance $x_j$ is classified by the target category $c$ and the corresponding embedded explanation $h_j$ is in the cluster $G_i$. If we take the largest element of $g_i$, say $g_{i,1}$, the corresponding base vector $w_1$ is related to the most important explanation of $x$ and therefore the corresponding
constraint $B(w_1, \theta_w)$ is the most significant for classifying $x_j$ as the target category $c$ and its weight is given by $g_{i1}$. Similarly, the second largest element of $g_i$, say $g_{i2}$, yields the next important constraint $B(w_2, \theta_w)$ with weight $g_{i2}$.

From these observations, we extract an explanation of the target category $c$ from each cluster. Consider a threshold $k_0$ to take important base vectors that affect significantly. We then take top $k_0$ base vectors and denote the set of these vectors by $S_i$. By combining all corresponding conditions of base vectors from $S_i$, we have a rectangle

$$B_{c,i} = \bigcap_{l \in S_i} B(w_l, \theta_w),$$

which is related to a underlying decision making constraint of $G_i$. Since there are $r_0$ clusters relevant to the target category $c$, by collecting all constraints we finally obtain a model explanation of the target category $c$

$$B_c = \bigcup_{i=1}^{r_0} B_{c,i}.$$  

Summarizing all the above observations, we formulate an optimization problem to find a model explanation of the target category $c$.

$$\max_{r, \theta_w, k_0} F_1(B_c),$$

$$\text{s.t. } r \leq r_{\text{max}}, \ k_0 \leq k$$

(6) The complexity of $B_c$ depends on the value of $r_0$ which is bounded by the value of $r$. For human understandable explanation, we consider a constraint $r_{\text{max}}$ of the number $r$ of clusters in our optimization problem as in (6). By solving (6) we have a model explanation of the target category $c$.

Solving the optimization problem for all categories we finally have a method to extract a model explanation $B$ of a classification model $f$. The maximum number of rectangles in each $B_c$ can be controlled by setting $r_{\text{max}}$. Also, by (6) the performance $F_1(B)$ is increasing in $r_{\text{max}}$. If the value of $r_{\text{max}}$ increases, a model explanation has the better performance, while it could be more complex since each $B_c$ has a chance to contain more rectangles. On other hand, if the value of $r_{\text{max}}$ decreases, the model explanation could be more simple, while the performance $F_1(B_c)$ decreases. Consequently, there is a tradeoff between the complexity of a model explanation and its performance, which is controlled by $r_{\text{max}}$.

Before closing this subsection, we would remark the generality of our approach. We have derived our method from the viewpoint of the local perturbation that introduced in (Ribeiro, Singh, and Guestrin 2016). However, in our method only a contribution matrix is required to extract the rule-like model explanation, which means the instance explanation step can be replaced with other techniques if we have well-defined interpretable space $\mathcal{F}$ and embedding $\mathcal{F}$.

### Experimental Results

This section provides experimental results of our proposed method to extract a model explanation. In our experiments we will use open datasets in the UCI machine learning repository (Lichman 2013). Moreover, to show the usability of our method in practice, we will also show the results by an industrial dataset, logs that diagnose the cause of disconnections in a long-term evolution (LTE) network. In the dataset, since more than one expert classifies the cause of disconnections, there is a guideline of the categories of causes to ensure the consistency. Consequently, we are able to check the validity of our method by comparing the guideline with our results. Because of the privacy issue, we will use a filtered dataset with obfuscation. So the dataset consists of 10 features whose names are $F_1, \cdots, F_{10}$, and 7 categories $C = \{C_1, \cdots, C_7\}$.

For our experiments, the random forest (RF), the adaptive boosting (AdaBoost) algorithm and the multi-layer perceptron (MLP) algorithm will be used for classification models. We choose these algorithms since they are the typical and basic representatives of ensemble, boosting, and deep learning that are widely used nowadays. Our implementation is based on the scikit-learn package (Pedregosa et al. 2011) to train classification models. Also, we will generate instance explanations with the help of the LIME (Ribeiro, Singh, and Guestrin 2016).

### Validation of Clustering Assumption

Our method has supposed that embedded explanations in the constraint space are well clustered. To validate this as-
Figure 3: Two dimensional t-SNE of embedded explanations. The category of an embedded explanation is indicated by a color. In most cases embedded explanations having the same category are well clustered into several groups.

Assumption, we evaluate the purity of each clusters with the K-means clustering. The results are plotted in Figure 2. The purity of a cluster is defined by the fraction of the number of embedded explanations in a cluster that are labeled by the most common category in the cluster. The y-axis of each graph in Figure 2 represents the purity over the clusters. The value of the purity is almost 1 in most datasets. Even though in the dermatology and LTE cases there are a few clusters having a low purity at $r = 3, 4$, both the median or the average are almost 1, which means most clusters have the high purity value. Hence, the embedded explanations are well concentrated, which validates our assumption in the previous section.

To see in details, we also visualize embedded explanations in Figure 3. We visualize the high-dimensional data by using t-SNE (Maaten and Hinton 2008). In fact, t-SNE distills the location of data points to embed into $\mathbb{R}^2$-plane, but it is still efficient to investigate the high-dimensional data. We use a model that trained by the random forest algorithm. Figure 3 shows the results of each dataset. The category of an instance is indicated by a color. In most cases, most embedded explanations having the same category are well clustered into several groups. Even the clusters are not clearly separated in the wine origin and dermatology datasets, it is still possible to find groups of embedded explanations.

**Explanation Extraction Results**

We now investigate an experiment to see the performance of a model explanation by our method. We here fix $k = 10$ in the NMF. We use 200 decision trees for the RF algorithm. The MLP is trained with two hidden layers of 128 nodes and the ReLU activations. Finally, the AdaBoost uses the logistic regression as its base estimator. In order to train a classification model, we split each dataset into training dataset (70%) and test dataset (30%).

Table 1 summarizes the performances $F_1(B)$ of model explanation. In the table, we use abbreviations. Wine: Wine origin dataset, Derm: Dermatology dataset, LTE: LTE network dataset. A value in the accuracy column is the accuracy of a trained model.

| Dataset | $C$ | $m$ | $N$ | Algorithm | Accuracy | $F_1$ |
|---------|-----|-----|-----|-----------|----------|-------|
| Wine    | 3   | 13  | 178 | RF        | 0.98     | 0.92  |
|         |     |     |     | MLP       | 0.97     | 0.94  |
|         |     |     |     | AdaBoost  | 0.99     | 0.92  |
| Derm    | 6   | 34  | 366 | RF        | 0.98     | 0.77  |
|         |     |     |     | MLP       | 0.96     | 0.72  |
|         |     |     |     | AdaBoost  | 0.97     | 0.87  |
| LTE     | 8   | 10  | 793 | RF        | 0.93     | 0.80  |
|         |     |     |     | MLP       | 0.91     | 0.83  |
|         |     |     |     | AdaBoost  | 0.86     | 0.79  |

**Table 2: Examples of extracted model explanations.**

| Category | Model Explanation |
|----------|-------------------|
| Wine.1   | $12.85 < \text{Alcohol} \& 755.0 < \text{Proline}$ |
| Wine.2   | $(\text{Color intensity} \leq 3.46 \& \text{Alcohol} \leq 12.85)$ or $(\text{Proline} \leq 682.5)$ |
| Wine.3   | $(3.82 < \text{Color inten.} \& \text{Flavanoids} \leq 0.975)$ |
| Derm.1   | $(0 < \text{Clubbing of rete ridges})$ or $(1 < \text{Thinning of suprapapillary epidermis} \& 0 < \text{Elongation of rete ridges})$ |
| Derm.2   | $\text{Koebner phenom.} \leq 0 \& 2 < \text{Spongiosis}$ & $\text{Disappearance of granular layer} \leq 0$ & $\text{& Clubbing of rete ridges} \leq 0$ & $\text{& Fibrosis of papillary dermis} \leq 0$ & $\text{& 0 < PNL infiltrate}$ |
| LTE.C1   | $0.05649 < F_1$ |
| LTE.C2   | $11 < F_9$ |
| LTE.C3   | $4 < F_{10} \& F_1 \leq 0.05649 \& 41 < F_6 \& F_0 \leq 11$ |
a model explanation rather than the accuracy. To verify the understandability of the results, we provide the examples of extracted model explanations in Table 2. In the table, due to the space limitation, we denote the category name with its dataset name. As seen in the table, the extracted explanation is intuitive. For instance, an extracted model explanation of the category setosa in the iris dataset is given by

\[ \text{petal width} \leq 3, \text{ petal length} \leq 1.6, \text{ sepal length} \leq 5.1, \]

from which we can directly understand that the setosa needs a short length of petal and that of sepal. In addition, the results in the LTE network dataset in Table 2 are almost the same with the guideline written by domain experts. Therefore, we conclude that our proposed method can extract a understandable explanation of a classification model.

We next perform experiments to see the effect of the parameter \( k \) in the NMF (Figure 4). We use the dermatology dataset since it has the larger number of categories and features than the other dataset has. As seen in the figure, the value of \( F_1(B) \) increases as the value of \( k \) increases. The parameter \( k \) is related to the number of base constraints that a classification model \( f \) has. If we have the fewer number of base constraints than the number of categories, the base constraints have a weak ability to distinguish categories. On the other hand, if \( k \) is large enough, base constraints have an ability to distinguish categories separately. Consequently, our proposed method needs to have a large enough value of \( k \) to extract a proper model explanation, which depends on the number of base constraints of a classification model.

**Conclusion**

In this paper, we propose an analytic method that extracts a understandable model explanation of a black-box classification model by using the nonnegative matrix factorization. We consider a concept of a model explanation in terms of the set approximation. We introduce a contribution matrix of features based on an interpretable explanation. By applying the NMF to the contribution matrix we extract the underlying decision making constraints of the classification model. Then combining those constraints we find a model explanation of the classification model. To see the validity of our approach, experiments are performed with open datasets as well as an industrial dataset. The experimental results show that our method extracts understandable and reasonable rule-like explanation of the model. So our method can help a service provider quickly decide the validity of a black-box model, and it can also help in reducing the development costs of a new intelligent service in practice.

**References**

[Bastani, Kim, and Bastani 2017] Bastani, O.; Kim, C.; and Bastani, H. 2017. Interpreting blackbox models via model extraction. arXiv:1705.08504.

[Fong and Vedaldi 2017] Fong, R., and Vedaldi, A. 2017. Interpretable explanations of black boxes by meaningful perturbation. arXiv:1704.03296.

[Kim, Rudin, and Shah 2014] Kim, B.; Rudin, C.; and Shah, J. A. 2014. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In Advances in Neural Information Processing Systems, 1952–1960.

[Kotsiantis, Zaharakis, and Pintelas 2007] Kotsiantis, S. B.; Zaharakis, I.; and Pintelas, P. 2007. Supervised machine learning: A review of classification techniques.

[Li and Belford 2002] Li, R.-H., and Belford, G. G. 2002. Instability of decision tree classification algorithms. In Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’02, 570–575. New York, NY, USA: ACM.

[Lichman 2013] Lichman, M. 2013. UCI machine learning repository.

[Lipton 2016] Lipton, Z. C. 2016. The mythos of model interpretability. In ICML Workshop on Human Interpretability of Machine Learning.

[Maaten and Hinton 2008] Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. Journal of Machine Learning Research 9(Nov):2579–2605.

[Núñez, Angulo, and Català 2002] Núñez, H.; Angulo, C.; and Català, A. 2002. Rule extraction from support vector machines. In Exaun, 107–112.

[Pedregosa et al. 2011] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12:2825–2830.

[Ribeiro, Singh, and Guestrin 2016] Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. Why should i trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1135–1144. ACM.

[Ross, Hughes, and Doshi-Velez 2017] Ross, A. S.; Hughes, M. C.; and Doshi-Velez, F. 2017. Right for the right reasons:
Training differentiable models by constraining their explanations.  

[Shrikumar, Greenside, and Kundaje 2017] Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences.  

[Simonyan, Vedaldi, and Zisserman 2014] Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. 2nd International Conference on Learning Representations Workshop.  

[Turner 2016] Turner, R. 2016. A model explanation system. In 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP), 1–6.  

[Witten et al. 2016] Witten, I. H.; Frank, E.; Hall, M. A.; and Pal, C. J. 2016. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.  

[Zintgraf et al. 2017] Zintgraf, L. M.; Cohen, T. S.; Adel, T.; and Welling, M. 2017. Visualizing deep neural network decisions: Prediction difference analysis. arXiv:1702.04595.