

Optimal Piecewise Approximations for Model Interpretation

Kartik Ahuja
Electrical and Computer Engineering
Department, UCLA
Los Angeles, CA 90095
Email: ahujak@ucla.edu

William Zame
Economics Department
UCLA
Los Angeles, CA 90095
Email: zame@econ.ucla.edu

Mihaela van der Schaar
Electrical and Computer Engineering
Department, UCLA
Los Angeles, CA 90095
Email: mihaela@ee.ucla.edu

Abstract—Recent literature interprets the predictions of “black-box” machine learning models (Neural Networks, Random Forests, etc.) by approximating these models in terms of simpler models such as piecewise linear or piecewise constant models. Existing literature does not provide guarantees on whether these approximations reflect the nature of the predictive model well, which can result in misleading interpretations. We provide a *tractable* dynamic programming algorithm that partitions the feature space into subsets and assigns a local model (constant/linear model) to provide piecewise constant/piecewise linear interpretations of an arbitrary predictive model. When approximation loss (between the interpretation and the predictive model) is measured in terms of mean squared error, our approximation is optimal; for more general loss functions, our interpretation is approximately optimal. Therefore, in both cases it probably approximately correctly (PAC) learns the predictive model. Experiments with real and synthetic data show that it provides significant improvements (in terms of mean squared error) over competing approaches. We also show real use cases to establish the utility of the proposed approach over competing approaches.

Index Terms—Interpretability, Dynamic Programming

I. INTRODUCTION

Machine Learning algorithms have proved extremely successful for a wide variety of supervised learning problems. However, in some domains, adoption of these algorithms has been hindered because the “black-box” nature of these algorithms makes their predictions difficult or impossible for potential users to interpret. This issue is especially important in the medical domain and security applications [1]. The European Union’s Law on Data Regulation taking effect in 2018 [2] makes it mandatory for “black-box” models to explain how they arrive at the predictions before implementing them in practice.

The problem of interpretation has received substantial attention in the literature recently [3] [4] [5] [6]. These papers have approached the problem of interpreting the black box model by approximating it with *piecewise models* (e.g., piecewise constant or piecewise linear) (See the justification in [7]).¹

Existing works on model interpretation that use piecewise models to approximate the global model [5] [3] often do not search the space of the piecewise models effectively and

can often result in poor approximations. As a result, the resulting interpretation may poorly reflect the true nature of the black-box. In general, searching for the best² piecewise approximation is non-trivial (justification provided later) and we address this problem in this work. Due to space limitation, we do not provide a detailed account of related works here and instead give them in the Supplementary Material [8].

Contribution In this paper we propose piecewise interpreter (PI), to interpret the black-box models. Our method is to use the black-box predictive model and a given data set to construct a partition of the feature space into subsets and to assign a simple local model to each subset. We propose a dynamic programming based approach to find the partition of the dataset and the set of local models. We prove that the output of PI is approximately optimal in several different cases, i.e. it PAC learns the predictive model. We use several real and synthetic datasets to show that the proposed approach results in significantly better interpretable model approximations compared to competing approaches.

In Figure 1, we show an example of a black-box model that predicts the risk of heart failure based on age and weight. We identify three phenotypes, where each phenotype corresponds to a risk interval, low risk, medium risk and high risk. In each interval, we use a fixed linear model to approximate the black-box model. The coefficients associated with age and weight vary significantly for the three phenotypes as shown. Further details can be explored in https://mlinterpreter.shinyapps.io/app_try/. The code for the proposed method can be found at <https://github.com/ahujak/Piecewise-Local-Linear-Model>.

II. PROBLEM FORMULATION

We are given a space \mathcal{X} of *features* and a space $\mathcal{Y} = [0, 1]^d$ of *labels*. We are given a predictive model $f : \mathcal{X} \rightarrow \mathcal{Y}$ (say a random forest based model or a deep neural network model). The data is distributed according to some *true distribution* \mathcal{D} (typically unknown) on \mathcal{X} . Our objective is to interpret f in terms of *interpretive models*, which are defined below. We seek to find a interpretive model that approximates f .

Interpretive models: The interpretive models we consider here represent the most commonly used models in literature on

¹We define piecewise models later.

²The notion of optimality is defined later.

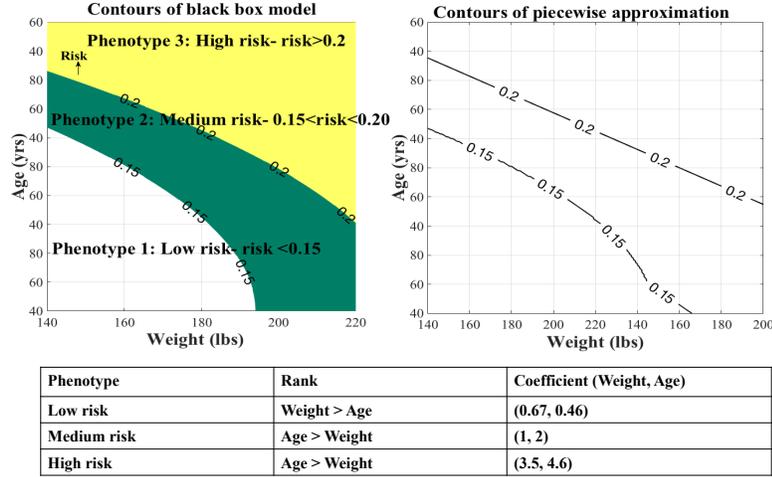


Fig. 1. Comparison of the black box model versus the piecewise approximation. White, Green and Yellow colored regions represent three different phenotypes. Coefficients of the features and the associated ranking is also shown.

model interpretation [3][7]. The interpretable models we consider are defined by partitioning \mathcal{X} into a finite number of disjoint sets and assigning a simple model (linear or constant model) to each set of the partition (In the Supplementary Material [8] we describe how many models in literature belong to this category). Hence, our notion of what it means to interpret locally coincides with the notions in [3][7].

Recall that a (finite) partition of a subset $A \subset \mathcal{X}$ is a family $\mathcal{Z} = \{Z_1, Z_2, \dots, Z_K\}$ of subsets of \mathcal{X} such that $\bigcup_{i=1}^K Z_i = A$ and $Z_i \cap Z_j = \emptyset$ if $i \neq j$. Given a partition \mathcal{Z} of A and a feature $a \in A$ we write $\mathcal{Z}(a)$ for the index of the unique element of the partition \mathcal{Z} to which a belongs. Write $\mathcal{P}(A)$ for the set of all (finite) partitions of A and $\mathcal{P}_K(A)$ for the set of partitions having at most K elements. We define $\mathcal{M} = \{M_1, \dots, M_K\}$ a set of local models, where model M_j corresponds to the local model for points in Z_j . Each local model M_j belongs to a set \mathcal{H} of models, where \mathcal{H} can be from the family of constant models ($M_j(x) = c$, where $c \in \mathbb{R}^d$) or linear models ($M_j(x) = Bx + c$, where $B \in \mathbb{R}^{d \times |\mathcal{X}|}$ and $c \in \mathbb{R}^d$).

Given a partition \mathcal{Z} of \mathcal{X} and the corresponding set of local models \mathcal{M} , we define an *interpretive model* $g_{\mathcal{M}, \mathcal{Z}} : \mathcal{X} \rightarrow Y$ by $g_{\mathcal{M}, \mathcal{Z}}(x) = M_{\mathcal{Z}(x)}(x)$.

Loss functions: We measure the goodness of fit of a proposed interpretation g for f in terms of a given *loss function*. We assume $\ell : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is in $\mathbb{C}^{2,3}$, strictly increasing such that $\ell(0) = 0$. We also assume that $\ell(|\cdot|)$ is strictly convex. We define the risk achieved by model $g_{\mathcal{M}, \mathcal{Z}}$ as follows

$$R(f, g_{\mathcal{M}, \mathcal{Z}}; \mathcal{D}) = E_{\mathcal{D}}[\ell(\|f(X) - g_{\mathcal{M}, \mathcal{Z}}(X)\|_s)] \quad (1)$$

where X is a feature from the distribution \mathcal{D} , the expectation is taken over the distribution \mathcal{D} , and $\|\cdot\|_s$ is the s -norm. Some examples of loss functions are squared function, i.e., $\ell(x) = x^2$ leading to standard mean squared error, $\ell(x) = x^4$.

³function, its derivative, and second derivative exist and are continuous

Risk Minimization: We impose an upper bound K on the number of sets in a partition (provided as input by the expert, for e.g., a clinician). Our objective is to find a partition \mathcal{Z} and a map $M : \mathcal{Z} \mapsto \mathcal{Y}$ (where each local model is drawn from \mathcal{H}) to minimize the true risk subject to the constraint that the size of the partition, i.e., $|\mathcal{Z}| \leq K$.

$$(\mathcal{M}^*, \mathcal{Z}^*) = \operatorname{argmin}_{\mathcal{M} \in \mathcal{H}^K, \mathcal{Z} \in \mathcal{P}_K(\mathcal{X})} R(f, g_{\mathcal{M}, \mathcal{Z}}; \mathcal{D}) \quad (2)$$

$g_{\mathcal{M}^*, \mathcal{Z}^*}$ is the best piecewise model that minimizes the above risk.

Empirical Risk Minimization: In practice, we do not know the true distribution \mathcal{D} so we cannot minimize the true risk; instead, we see only a finite dataset (training set) $D = \{x_i\}_{i=1}^N$ drawn from the true distribution. For given \mathcal{M} , \mathcal{Z} the empirical risk is

$$\hat{R}(\mathcal{M}, \mathcal{Z}; D) = \frac{1}{n} \sum_{x_i \in D} [\ell(\|f(x_i) - g_{\mathcal{M}, \mathcal{Z}}(x_i)\|_s)] \quad (3)$$

The spirit of Probably Approximately Correct (PAC) learning [9] suggests that we should minimize the empirical risk:

$$(\mathcal{M}^\dagger, \mathcal{Z}^\dagger) = \operatorname{argmin}_{\mathcal{M} \in \mathcal{H}^K, \mathcal{Z} \in \mathcal{P}_K(\mathcal{X})} \hat{R}(\mathcal{M}, \mathcal{Z}; D) \quad (4)$$

Later we will show that solving the above empirical risk minimization problem is PAC solution to the actual risk minimization problem in (2). We cannot solve the above problem using brute force search because it requires searching among $\mathcal{O}(|D|^K)$ partitions, which becomes intractable very quickly with increase in $|D|$ and K . In the next section, we propose an Algorithm to solve the above problem.

III. PIECEWISE INTERPRETER ALGORITHM

In this section, we develop the Piecewise Interpreter Algorithm to solve the problem discussed above. Without loss of generality we assume that all the data points x_i in D are sorted in the increasing order of the norm of $f(x_i)$, i.e. $\|f(x_i)\|$.

We give a summary of the working of the Algorithm next. We give a detailed analysis of the Algorithm in Section IV and V. There are two parts to the Algorithm (Algorithm 1 and 2). In the first part, the Algorithm partitions the data D into subsets and finds an optimal local model corresponding to each subset. The division of the dataset into these subsets relies on dynamic programming. Suppose that the Algorithm wants to divide the first p points into q subsets. Also, suppose that the Algorithm has already constructed a partition to divide the first m points into k subsets for all $m \leq p - 1$ and for all $k \leq q - 1$. The risk achieved by partition of m points where the size of the partition is k is defined as $V(m, k)$. For each $x_i \in D, x_j \in D$, where $i \leq j$, define a subset of the data as follows.

$$D(i, j) = \{x : x \in D \ \& \ \|f(x_i)\| \leq \|f(x)\| \leq \|f(x_j)\|\}$$

We define the optimal local model on $D(i, j)$ as follows

$$G(i, j) = \min_{h \in \mathcal{H}} \frac{1}{|D(i, j)|} \sum_{x_r \in D(i, j)} [\ell(\|f(x_r) - h(x_r)\|_s)]$$

$$M'(i, j) = \operatorname{argmin}_{h \in \mathcal{H}} \frac{1}{|D(i, j)|} \sum_{x_r \in D(i, j)} [\ell(\|f(x_r) - h(x_r)\|_s)]$$

The Algorithm constructs a partition to divide p points into q subsets as follows

$$V(p, q) = \min_{n' \in \{1, \dots, p-1\}} [V(n', q-1) + G(n'+1, p)]$$

$$\Phi(p, q) = \operatorname{argmin}_{n' \in \{1, \dots, p-1\}} [V(n', q-1) + G(n'+1, p)]$$

where $\Phi(p, q)$ is the index of the first data point in the q^{th} subset. The subset q consists of all the points indexed $\{\Phi(p, q), \dots, p\}$. The model for the q^{th} subset is $M'(\Phi(p, q) + 1, p)$. Similarly, the next subset, i.e., the $(q+1)^{\text{th}}$ subset can be computed recursively as $\{\Phi(\Phi(p, q), q-1), \dots, \Phi(p, q) - 1\}$ and so on. In the first part of the Algorithm, we construct a partition of D and the corresponding set of local models. In the second part of the Algorithm, we extend this partition from the dataset D to the set \mathcal{X} . We write the function that is output by the Algorithm 2 as $g_{\mathcal{M}^\#, \mathcal{Z}^\#}$.

IV. MAIN RESULTS

Our goal in this section is to show that the output of the Algorithm PAC learns f and the Algorithm computes $g_{\mathcal{M}^\#, \mathcal{Z}^\#}$ in polynomial time. Before we state the results we state certain assumptions.

Assumption 1: We assume that family of local models \mathcal{H} corresponds to the set of constant models, i.e. the interpretive models are piecewise constant.

Assumption 2: We also assume that if $\|f(x_i)\| < \|f(x_j)\| \implies f(x_i) < f(x_j)$, i.e., $f(x_j)$ Pareto dominates

$f(x_i)$. If f is scalar, i.e., $f(x) \in [0, 1], \forall x \in \mathcal{X}$, then the above assumption always holds.

Complexity of Algorithms 1 and 2 We begin by deriving the computational complexity of the Algorithm. For the complexity results, we assume that the loss function is the mean squared error and we assume that the local models are drawn from the constant model class.

Theorem 1. If Assumption 1 holds and the loss function is mean squared error, then the computational complexity of Algorithm 1 and 2 together is $\mathcal{O}(|D|^3 K d)$.

The Proof of Theorem 1 is given in the Supplementary Material [8]. While $\mathcal{O}(|D|^3 K d)$ is much better than the brute force search among $\approx |D|^K$ ordered partitions, it can still be large. Hence, we propose some speed-ups (in the Supplementary Material [8]) and used them in the Numerical Results Section.

PAC learnability of Algorithms 1 and 2: In this section, we discuss whether the outcome of Algorithm 1 and 2, i.e. $g_{\mathcal{M}^\#, \mathcal{Z}^\#}$ PAC learns f .

For the next theorem, let Assumption 1 and 2 be true. In the next section, we discuss the optimality for piecewise linear models.

Theorem 2 If Assumption 1 and 2 hold, then $\forall \epsilon > 0, \delta \in (0, 1), \exists n^*(\epsilon, \delta)$ such that if D is drawn i.i.d. from \mathcal{D} and $|D| \geq n^*(\epsilon, \delta)$, then with probability at least $1 - \delta$,

$$|R(f, g_{\mathcal{M}^\#, \mathcal{Z}^\#}; \mathcal{D}) - R(f, g_{\mathcal{M}^*, \mathcal{Z}^*}; \mathcal{D})| \leq \epsilon$$

The proof of Theorem 2 is given in the Supplementary Material [8].

A. Piecewise linear models

In Theorem 2, we assumed that the local model is constant. In this section, we discuss Theorem 2 in the context of piecewise linear models. We divide the discussion into two parts.

First, suppose that we use a linear model as the local model in Algorithm 1 and 2, then it is hard to show PAC learnability as in Theorem 2. However, if we restrict the set of partitions we search to a special class of partitions, i.e., ordered partitions, that we define in Section V A, then we can extend Theorem 2 to this setting as well (See the Piecewise Linear Models Section in Supplementary Material [8]).

Second, in many cases, we only require that the interpretation to be able to predict the importance that a model associates with different features, i.e., the gradient of the black-box w.r.t the features. Suppose that the outcome of the black-box is a scalar, i.e., $d = 1$ and suppose that the black-box model f is differentiable w.r.t x almost everywhere. In this case, instead of finding a piecewise linear interpretive model that minimizes the distance w.r.t f , we can find an interpretive model that minimizes the distance w.r.t the gradient of f , i.e., ∇f . We know that the gradient of a piecewise linear function is defined almost everywhere and corresponds to a piecewise constant function. Hence, we can use a piecewise constant model that explains ∇f . Therefore, we can apply Theorem 2 and conclude that the interpretive model PAC learns the gradient of f .

We summarize the results for the piecewise constant function and piecewise linear models in Table 1.

Algorithm 1 Computing value and index functions

Input: Dataset D , Number of subsets K

Initialize: Define $V'(1, k) = 0, \forall k \in \{1, \dots, K\}$.

For each $x_i \in D, x_j \in D$ such that $i \leq j$, define $D(i, j) = \{x : x \in D \text{ and } \|f(x_i)\| \leq \|f(x)\| \leq \|f(x_j)\|\}$

$G(i, j) = \min_{h \in \mathcal{H}} \frac{1}{|D(i, j)|} \sum_{x_r \in D(i, j)} [\ell(\|f(x_r) - h(x_r)\|_s)]$

$M'(i, j) = \arg \min_{h \in \mathcal{H}} \frac{1}{|D(i, j)|} \sum_{x_r \in D(i, j)} [\ell(\|f(x_r) - h(x_r)\|_s)]$

for $n \in \{2, \dots, |D|\}$ **do**
 for $k \in \{1, \dots, K\}$ **do**

$$V'(n, k) = \min_{n' \in \{1, \dots, n-1\}} [V'(n', k-1) + G(n'+1, n)] \quad (5)$$

$$\Phi(n, k) = \operatorname{argmin}_{n' \in \{1, \dots, n-1\}} [V'(n', k-1) + G(n'+1, n)] \quad (6)$$

Output: Value function V' , Index function Φ

Algorithm 2 Computing partitions using the index function

1: **Input:** Index function Φ , Black box predictive model f

2: **Initialization:** $h_u = |D|$

3: **for** $k \in \{1, \dots, K\}$ **do**

4: $h_l = \Phi(h_u, K - k + 1)$

5: $Z_{K-k+1} = \{x : \|f(x_{h_l})\| < \|f(x)\| \leq \|f(x_{h_u})\|\}$

6: $M_{K-k+1} = M'(h_l + 1, h_u)$

7: $h_u = h_l$

8:

9: **Output:** $\mathcal{Z}^\# = \{Z_1, \dots, Z_K\}$,

10: $\mathcal{M}^\# = \{M_1, \dots, M_K\}$

V. PRINCIPLES UNDERLYING THE MAIN RESULTS AND THE ALGORITHM

In this section, we first describe the principles behind the construction of the Algorithm and then we develop the propositions that lead up to the main result discussed in Section IV. In Section V A, we identify a new class of partitions that we call *ordered partitions* and restrict our search in that space. The empirical risk function defined earlier follows Bellman principle when we restrict the search space. In Section V B, we show that the dynamic programming approach presented in Algorithm 1 and 2 finds the optimal ordered partition. In Section V C, we show that the risk achieved by the optimal ordered partition and the corresponding models found by the proposed algorithm is very close to the risk achieved by the optimal partition $\mathcal{M}^\dagger, \mathcal{Z}^\dagger$ defined in (4). We then use results on PAC learning [9] and certain properties of the family of piecewise functions to establish the main result Theorem 2.

A. Ordered Partitions

We say that the partition \mathcal{Z} of $A \subset \mathcal{X}$ is *ordered* if for every $Z, Z' \in \mathcal{Z}$ with $Z \neq Z'$, either

(i) for all $z \in Z, z' \in Z'$ we have $\|f(z)\| < \|f(z')\|$, or

(ii) for all $z \in Z, z' \in Z'$ we have $\|f(z)\| > \|f(z')\|$

Write $\mathcal{P}^*(A)$ for the set of all ordered partitions of A and $\mathcal{P}_K^*(A)$ for the set of all ordered partitions of A having at most

K (non-empty) elements. We now show that the risk defined over the partitions follows the Bellman principle. Let \mathcal{Z} be a partition of $A \subset \mathcal{X}$ and let $\mathcal{Z}', \mathcal{Z}'' \subset \mathcal{Z}$ be a partition of \mathcal{Z} ; i.e. $\mathcal{Z}' \cup \mathcal{Z}'' = \mathcal{Z}$ and $\mathcal{Z}' \cap \mathcal{Z}'' = \emptyset$.

Bellman Principle Let \mathcal{Z} be an ordered partition of \mathcal{X} , \mathcal{M} be the set of optimal local models, and assume that $\hat{R}(\mathcal{M}, \mathcal{Z}; D)$ minimizes the risk among all ordered partitions of \mathcal{X} with at most $|\mathcal{Z}|$ elements. If $\mathcal{Z}', \mathcal{Z}'' \subset \mathcal{Z}$ is a partition of \mathcal{Z} , $\mathcal{M}', \mathcal{M}'' \subset \mathcal{M}$ is a partition of \mathcal{M} , then $\hat{R}(\mathcal{M}', \mathcal{Z}'; D)$ minimizes the risk among all ordered partitions of A' with at most $|\mathcal{Z}'|$ elements. If this were not true then we could find another ordered partition \mathcal{Z}^* of A' with lower risk. But then $\mathcal{Z}^* \cup \mathcal{Z}''$ would be an ordered partition of \mathcal{X} with lower risk than \mathcal{Z} , which would be a contradiction.

B. Optimal Ordered Partitions

In this section, we will show that the Algorithm conducts a search in the space of ordered partitions and finds the solution that minimizes the objective in (4).

Proposition 1. The output of the Algorithm 1 and 2 achieves a risk value equal to $\min_{\mathcal{M} \in \mathcal{H}^K, \mathcal{Z} \in \mathcal{P}_K^*(\mathcal{X})} \hat{R}(\mathcal{M}, \mathcal{Z}; D)$.

The proof of Proposition 1 is given in the Supplementary Material [8]. We give a brief proof sketch next.

Proof Sketch 1. In the first part, the Algorithm constructs a value function and an index function; the second part uses the value function and the index function to produce an ordered partition. We use induction and Bellman principle. Suppose we consider the first p points and we want to partition them into q subsets. Suppose that the optimal partitions for the first m points into k subsets is known $\forall m \leq p-1$ and $\forall k \leq q-1$. From Bellman principle, we know that the risk achieved by the optimal partition of p points into q subsets is equal to $\min_{n' \in \{1, \dots, m\}} [V(n', q-1) + G(n'+1, p)]$. From the construction of the Algorithm it follows that the partition of p points into q subsets computed by the Algorithm is optimal because the risk achieved by it is equal to $\min_{n' \in \{1, \dots, m\}} [V(n', q-1) + G(n'+1, p)]$. QED

TABLE I
SUMMARY OF RESULTS ON PAC LEARNABILITY FOR DIFFERENT SETTINGS

Interpretive Model	Set of Partitions	Shape-penalty	Risk function	Output
Piecewise constant	All & size $\leq K$	No	$E[l(\ f - g_{\mathcal{M}, \mathcal{Z}}\)]$	PAC learns f
Piecewise linear	Ordered & size $\leq K$	No	$E[l(\ f - g_{\mathcal{M}, \mathcal{Z}}\)]$	PAC learns f
Piecewise linear	All & size $\leq K$	No	$E[l(\ \nabla f - \nabla g_{\mathcal{M}, \mathcal{Z}}\)]$	PAC learns ∇f
Piecewise constant/linear	Ordered & size $\leq K$	Yes	$E[l(\ f - g_{\mathcal{M}, \mathcal{Z}}\) + \text{penalty}]$	PAC learns f

C. Optimal Ordered Partitions are Optimal Among all Partitions

We have shown that the ordered partitions constructed in Algorithms 1 and 2 are optimal among ordered partitions; we justify our focus on ordered partitions by demonstrating their optimality *among all partitions*. We invoke Assumptions 1 and 2 from Section IV for the next two propositions.

Proposition 2. If Assumption 1 and 2 hold and the loss function is mean squared error then $\min_{\mathcal{M}, \mathcal{Z} \in \mathcal{P}_K(\mathcal{X})} \hat{R}(\mathcal{M}, \mathcal{Z}; D) = \min_{\mathcal{M}, \mathcal{Z} \in \mathcal{P}_K^*(\mathcal{X})} \hat{R}(\mathcal{M}, \mathcal{Z}; D)$

The proof of Proposition 2 is in the Supplementary Material [8]. In the next Proposition, we consider more general loss functions unlike the Proposition 2.

In the general case in which the loss function is only strictly convex (e.g., $\ell(x) = x^4$), we show that the risk achieved by optimal ordered partitions is *probably approximately* the minimal risk over arbitrary partitions.

Proposition 3. If Assumption 1 and 2 hold, then for every $\epsilon, \delta > 0$ and every K there is some $m^*(\epsilon, \delta, K)$ such that if the training set D is drawn i.i.d. from the distribution \mathcal{D} and $|D| \geq m^*(\epsilon, \delta, K)$, then with probability at least $1 - \delta$ we have $|\min_{\mathcal{M}, \mathcal{Z} \in \mathcal{P}_K(\mathcal{X})} R(\mathcal{M}, \mathcal{Z}; D) - \min_{\mathcal{M}, \mathcal{Z} \in \mathcal{P}_K^*(\mathcal{X})} R(\mathcal{M}, \mathcal{Z}; D)| < \epsilon$

The proof of Proposition 3 is in the Supplementary Material [8]. We give a brief Proof Sketch here.

Proof Sketch 2. We first identify a class of partitions that we call “dense partitions” that satisfy the following property. As the total number of points grows to infinity, the number of points in reach region of the partition should also diverge. We show that instead of searching among all the partitions it is sufficient to search among the dense partitions. Next, we want to show that if a partition is dense and not ordered, then the loss function can always be improved by making it ordered. To show this, we need to keep a track of the change in the loss function as we move from an unordered to an ordered partition. This is non-trivial to do because the loss function is general and the solution to (4) for a fixed partition does not have a closed form. To track this change, we use *influence functions* [10] and show that the sign of change is always non-positive. QED

We have now demonstrated that the output of Algorithm 1 and 2 approximately minimize the empirical risk defined in (4). In the Supplementary Material [8] we show that the

hypothesis space represented by the piecewise constant/linear family, which consists of infinite elements, can be approximated with a finite hypothesis class. Next, we use results on PAC learning in [9] to prove Theorem 2.

VI. EXTENSIONS

A. Constraint on the shape of the regions:

In the previous section, we developed a method to partition \mathcal{X} . However, the subsets forming the partition \mathcal{Z} such as Z_j might be hard to characterize. For instance Z_j may not be a hypercube or even a polyhedron. We add a constraint to the optimization problem to ensure that each of the regions in the partition is easier to characterize.

We require that for each region Z_i all the points in that region are closer to the centroid of Z_i than to the centroid of any other region. Adding such a constraint ensures that each region in the partition is easily described in terms of the centroids. Moreover, it also ensures that each region is a polyhedron.

Suppose data point $x_i \in Z_j$. Let the centroid of the region j is denoted as μ_{Z_j} , where $\mu_{Z_j} = \frac{1}{|Z_j|} \sum_{x_k \in Z_j} x_k$.

$\|x_i - \mu_{Z_j}\| \leq \|x_i - \mu_{Z_k}\|, \forall k \neq j$. We add this constraint as a penalty term to the empirical risk defined in (3). The penalized expression for the risk is given as

$$\hat{R}(\mathcal{M}, \mathcal{Z}; D) + \sum_{j,k} \lambda \sum_{x_i \in Z_j} (\|x_i - \mu_{Z_j}\| - \|x_i - \mu_{Z_k}\|)$$

where λ is the price of not satisfying the constraint.

We can modify the Algorithm 1 and 2 to incorporate the penalty terms (See details in the Supplementary Material [8]). We can also show that Theorem 1 and 2 can be adapted to this setting provided the space of the partitions is restricted to ordered partitions (See the details in the Supplementary Material [8]). We provide Numerical Results corresponding to this setting in the Supplementary Materials [8]. We provide a summary of all the results in Table I.

VII. RESULTS

A. Performance comparison

In this section, we describe the numerical results from experiments using both synthetic and real datasets. We carry out two types of comparisons: one using constant local models and the other using linear local models. We refer to our approach as the piecewise interpreter PI when we use constant local models and PI-L when we use linear local models. The experiments were conducted in Python.

TABLE II
COMPARISON OF PI WITH BENCHMARKS IN TERMS OF MSE FOR PIECEWISE CONSTANT MODELS. BLACK BOX MODEL: NEURAL NETWORK

Algorithm	Synthetic		Pollution		Maggic		Wine	
	In	Out	In	Out	In	Out	In	Out
PI	5	11	12	21	8	18	4	8
EQ	19	32	30	31	30	48	8	17
UD	24	25	40	44	30	35	15	15
RT	196	259	126	131	130	210	34	48

TABLE III
COMPARISON OF PI WITH BENCHMARKS IN TERMS OF MSE FOR PIECEWISE LINEAR MODELS. BLACK BOX MODEL: NEURAL NETWORK

Algorithm	Synthetic		Pollution		Maggic		Wine	
	In	Out	In	Out	In	Out	In	Out
PI-L	1	15	5	30	1.9	26	5.7	40
EQ-L	5	19	20	30	8	31	7.3	42
UD-L	4	18	0.21	27	6.4	570	11	120
RT	196	259	126	131	130	210	34	48

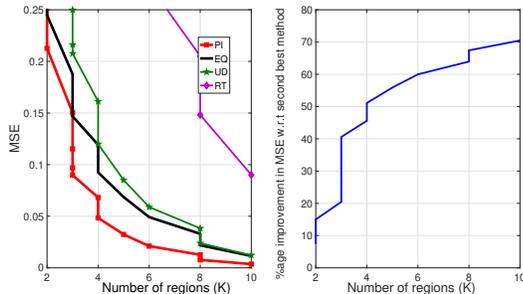


Fig. 2. MSE as a function of the number of regions (K) in the piecewise approximation

Benchmarks for piecewise constant models: Three natural approaches to partitioning are described next. **Equal Quantiles (EQ)** Divide the feature space into K intervals so that the K corresponding populations are equal. **Uniform Division (UD)** Divide the label space (the range of risk values) into K intervals of equal length and partition the feature space into the corresponding intervals. **Regression Tree (RT)** Use the predictive model f to construct a regression tree that approximates f with K leaves.

Benchmarks for piecewise linear models If we use linear models inside each region, then we also need to change the benchmarks for fair comparisons. Therefore, we use the above approaches UD and EQ to partition the feature space and then within each cluster that is identified we use local-linear model to learn a local model. We compare with the combination of UD with a local-linear model (EQ with local-linear model) which we refer to as UD-L (EQ-L).

Performance metric and comparisons We carry out both in sample and out of sample comparisons. We use mean squared error (MSE) as the loss. For every dataset, we do five-fold cross-validation (80 percent data for training and 20 percent data for testing). In the following comparisons, we set $K = 10$ (a maximum of ten clusters is allowed). In addition, we make

sure that all the methods use the same number of clusters so that we have fair comparisons. Also, since the complexity of the method is cubic in the size of the dataset, we use uniform sampling with a rate of 20 samples per iteration (See the Supplementary Materials [8] for details). This procedure works for our largest dataset with around 40,000 patients and 30 feature dimensions.

Black Box Models We use two black box models: a feedforward fully connected neural network (NN) (3 layers and 200 nodes per layer) and Random Forest Regression (10 trees, tree depth is 20).

Datasets. We use one synthetic dataset and three real datasets. Two of the real datasets are public datasets from UCI repository. The third dataset is the mortality prediction dataset (Maggic dataset) from [11]. The details of the datasets are in the Supplementary Materials [8].

Comparisons For every dataset (synthetic and real), we first learn the black box models above and then compare the performance of our PI with that of other approaches, being careful to compare piecewise constant interpretations with piecewise constant interpretations and piecewise linear interpretations with piecewise linear interpretations. Tables II,III, show the comparisons for the NN model; the comparisons for the RF model are similar and are shown in the Supplementary Materials [8]. Table II shows the comparisons for piecewise constant interpreters; Table III shows the comparisons for piecewise linear interpreters. In both tables, we see that in most settings our approach performs much better than other methods (both in and out of sample).

In Table II,III, we compared the MSE between the interpretive model's prediction and the black box. In the Supplementary Material [8], we show the MSE between the gradient of a piecewise linear model and the gradient of the black-box model. In the Supplementary Material [8], we also show the performance of the PI under the additional shape constraints.

Impact of choice of number of regions K : In Figure 2, we compare the performance of the proposed method with other approaches by varying the maximum number of regions from 2-10. Again, our approach consistently outperforms other methods regardless of the choice of K .

B. Phenotypes

In this subsection, we compare some of the phenotypes (Each region in the partition corresponds to a phenotype. Hence, a subject belongs to a certain phenotype if its feature values belong to the region corresponding to it.) that are computed by the PI for the Maggic dataset. The complete tables describing the phenotypes are in Supplementary Material [8]. Phenotype 2 consists of patients who have poor circulatory systems (exemplified by a low ejection fraction) but normal renal function; for these patients, the most predictive features are blood pressure, myocardial infarction, and diabetes, which are related to circulatory problems. Phenotype 8 consists of patients who have poor circulatory function and poor renal function; for these patients, the most predictive features are creatinine, hypertension, and blood pressure, which are related to both the renal and circulatory systems. As we can see from this example, an understanding of the most discriminative and most predictive features for each phenotype can help the clinician to better understand both the “how” and the “why” of the predictions of the underlying model.

VIII. CONCLUSION

This paper provides a novel way to construct piecewise approximations of a black box model. Our approach uses dynamic programming to partition the feature space into regions and then assigns a simple local model within each region. When we require the local models to be constant (linear) on each cluster, our piecewise approximation is optimal (approximately optimal), i.e. it achieves the minimum loss, where the loss measures the distance between the approximation and the actual prediction. We carry out experiments show that the proposed approach achieves a smaller loss and better reflects the black box model compared to other approaches.

IX. ACKNOWLEDGEMENT

We would like to thank Prof. Gregory J. Pottie (Electrical and Computer Engineering, UCLA) for his comments on the manuscript. We would like to acknowledge the Office of Naval Research (ONR) and the National Science Foundation (NSF) Award 1524417 for supporting this work. Kartik Ahuja would also like to thank Guru Krupa Fellowship for the support.

REFERENCES

- [1] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, “Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission,” in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 1721–1730.
- [2] B. Goodman and S. Flaxman, “European union regulations on algorithmic decision-making and a” right to explanation”,” *arXiv preprint arXiv:1606.08813*, 2016.
- [3] M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should i trust you?: Explaining the predictions of any classifier,” in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 2016, pp. 1135–1144.
- [4] A. Shrikumar, P. Greenside, and A. Kundaje, “Learning important features through propagating activation differences,” *arXiv preprint arXiv:1704.02685*, 2017.
- [5] O. Bastani, C. Kim, and H. Bastani, “Interpreting blackbox models via model extraction,” *arXiv preprint arXiv:1705.08504*, 2017.
- [6] J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan, “Learning to explain: An information-theoretic perspective on model interpretation,” *arXiv preprint arXiv:1802.07814*, 2018.
- [7] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems*, 2017, pp. 4765–4774.
- [8] K. Ahuja, W. Zame, and M. Schaar, “Supplementary materials for optimal piecewise approximations for model interpretation.” [Online]. Available: http://www.vanderschaar-lab.com/papers/Supplementary_Material_for_Optimal_Piecewise_Approximations.pdf
- [9] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [10] R. D. Cook and S. Weisberg, *Residuals and influence in regression*. New York: Chapman and Hall, 1982.
- [11] S. J. Pocock, C. A. Ariti, J. J. McMurray, A. Maggioni, L. Køber, I. B. Squire, K. Swedberg, J. Dobson, K. K. Poppe, G. A. Whalley *et al.*, “Predicting survival in heart failure: a risk score based on 39 372 patients from 30 studies,” *European heart journal*, vol. 34, no. 19, pp. 1404–1413, 2012.