Equi-explanation Maps: Concise and Informative Global Summary Explanations

Tanya Chowdhury
University of Massachusetts Amherst
Massachusetts, USA

Razieh Rahimi
University of Massachusetts Amherst
Massachusetts, USA

James Allan
University of Massachusetts Amherst
Massachusetts, USA

ABSTRACT

We attempt to summarize the model logic of a black-box classification model in order to generate concise and informative global explanations. We propose equi-explanation maps, a new explanation data-structure that presents the region of interest as a union of equi-explanation subspaces along with their explanation vectors. We then propose E-Map, a method to generate equi-explanation maps. We demonstrate the broad utility of our approach by generating equi-explanation maps for various binary classification models (Logistic Regression, SVM, MLP, and XGBoost) on the UCI Heart disease dataset and the Pima Indians diabetes dataset. Each subspace in our generated map is the union of d-dimensional hyper-cuboids which can be compactly represented for the sake of interpretability. For each of these subspaces, we present linear explanations assigning a weight to each explanation feature. We justify the use of equi-explanation maps in comparison to other global explanation methods by evaluating in terms of *interpretability*, *fidelity*, and *informativeness*. A user study further corroborates the use of equi-explanation maps to generate compact and informative global explanations.

KEYWORDS

explainability, subspace interpretability, global explanations, explaining classifiers, model-logic subspaces

ACM Reference Format:
Tanya Chowdhury, Razieh Rahimi, and James Allan. 2022. Equi-explanation Maps: Concise and Informative Global Summary Explanations. In *2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22), June 21–24, 2022, Seoul, Republic of Korea.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3531146.3533112

1 INTRODUCTION

Wikipedia defines Explainable AI as *AI in which the results of the solution can be understood by humans*. Most models today accept a set of features (tabular or categorical) and combine them in a carefully constructed though often obscure way to produce a result. An "explanation" uses the same or different features to generate simple, interpretable information that gives an insight on how the model might have arrived at that result. For example, a complex neural model might be explained by a linear combination of a subset of the features. As machine learning models are increasingly being used in real-world decision making, it is important to provide explanations of model predictions to guide their use and to improve understanding of them.

Explanation algorithms which explain a single model prediction are known as *local* explanation algorithms, while those which approximate characteristics of an entire model are known as *global* explanation algorithms. Explanation algorithms which explain predictions by taking into consideration the original model parameters are known as *model-introspective* explainers. Methods which treat the original models as a black-box, only to learn model characteristics using secondary training data, are known as *model-agnostic* explainers. Algorithms in which we generate explanations for a model after it has been trained are known as *post-hoc* explanation algorithms. Explanation algorithms differ in *basic units* of explanation: some methods use the features as-is for interpretability, while some use mappings of features for the same. Different interpretability methods map the *interaction between features* to various degrees. Most explainability methods produce linear interpretations of model predictions, thus ignoring all inter-feature interaction terms. Generating explanations is also *time* and *resource* dependent. Some algorithms assign a time *budget*, and return the optimal explanation model derived within the assigned budget.

In this work, we focus on model-agnostic post-hoc linear interpretability techniques. The problem of *how the model logic varies across the input space* has not been studied well. Assume that a medical practitioner has to rely on an ML model decision to choose between different treatment plans for heart patients. Before relying on the model for such a critical decision, they would like a system to summarize the *basis* on which the model makes decisions for different values of patient statistics (e.g., *smoking* and *exercise*) [6]. Existing global linear explanation methods [12, 15] at best return a set of representative instances which cannot be used to give answers to such questions.

To generate more informative global explanations, we propose dividing the desired region of explanation features into *subspaces* based on similar logic, i.e. ϵ*-equi-explanation* subspaces (Figure 1). In this work, we focus on the task of binary classification.

Each ϵ-equi-explanation subspace is a union of non-overlapping hyper-cuboids, each hyper-cuboid representing the range of values it covers over the explanation features. We employ a divide-and-conquer based approach, where in each step of the memoized recursion function, we compute the explanation vector for each vertex of the obtained hyper-cuboid. Explanation vectors for instances are computed based on their nearest Decision Boundary Point (DBP), a method which approximates LIME results [12] but with less uncertainty. We finally generate a linear explanation for

FAccT '22, June 21–24, 2022, Seoul, Republic of Korea

ACM ISBN 978-1-4503-9352-2/22/06…$15.00
https://doi.org/10.1145/3531146.3533112

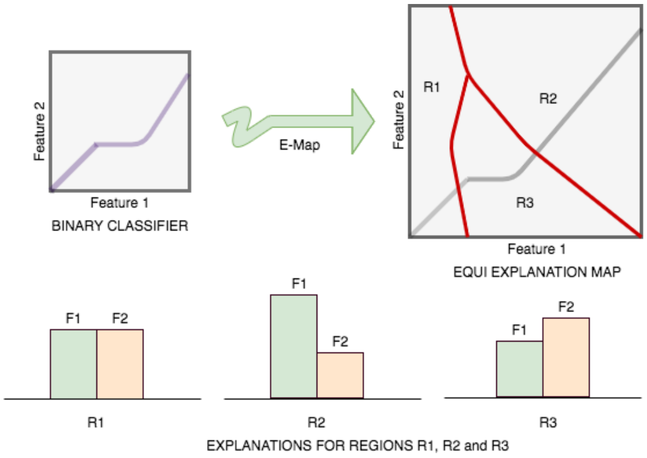

Figure 1: A hypothetical binary classifier (top left) partitioned into equi-explanation maps using the proposed E-Map algorithm. The algorithm partitions the feature space (here containing 2 features f1 and f2) into three regions (R1,R2 and R3) based on similar model logic. Bar charts in the bottom row represent relative importance of the features in each of the 3 regions.

each equi-explanation region by aggregating its member hypercuboid explanation vectors based on weight.

In order to study a given black-box model, our algorithm delivers compact and informative global explanations, presenting a set of equi-explanation regions and their corresponding explanation vectors. We adapt relevant global explanation methods such as SP-LIME [12], Guided-LIME [15], SHAP [10, 11], and MUSE [6] to form strong baselines and justify the use of equi-explanation maps on grounds of *Interpretability*, *Fidelity*, and *Informativeness*. We also conduct a user study to demonstrate the effectiveness of our new explanation format in comparison to a strong non-linear global explanation method [6] and show that equi-explanation maps outperform the strongest baseline considerably in terms of Informativeness.

In the spirit of reproducibility, our implementation is available here[1].

Our main contributions are as follows:

- We propose equi-explanation maps: a concise yet informative new data structure to summarize the model logic of a black-box classifier in order to generate concise and informative global explanations. In doing so we propose the task of *Global Summary Explanation Generation.*
- We propose the E-Map algorithm : a divide an conquer based architecture - specifically to generate the equi-explanation map explanation format.
- We propose new metrics to uniformly compare E-Map generated equi-explanation maps with other linear, additive global explanation methods. We also conduct a user study to prove the effectiveness of our proposed explanation format with respect to mimic model-based global explanation methods.

2 BACKGROUND

LIME [12] is a popular local explanation model in the machine learning literature. It is a model-agnostic linear explanation method that locally approximates a classifier with an interpretable model. It does so by perturbing inputs in a locality near the instance of interest, and generating labels for the perturbed inputs using the original model. Let $\mathcal{G}$ be the class of interpretable models. The explanation model obtained by LIME, for an instance x is:

$$E(x) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in Z} \pi_x(z)(f(z) - g(z'))^2 \tag{2}$$

$$\pi_x(z) = \exp(-D(x, z)^2/\sigma^2), \tag{3}$$

where $\Omega(g)$ represents the complexity of the learnt explanation model, π_x describes the locality around x (usually represented by an exponential kernel on a distance function), and $\mathcal{L}(f, g, \pi_x)$ is a model of how unfaithful the explanation model g is in neighborhood of x. D denotes a distance function (cosine in this case). In order to learn the local behavior of f around x, they sample points uniformly at random in the proximity of x. These samples are generated by perturbing the original input, and replacing some feature values by zero. Rebeiro et al. [12] also propose SP-LIME, a method that selects a set of representative instances for global interpretability via submodular optimization.

Laugel et al. [8] improve on LIME's sampling techniques to optimize model fidelity. They generate surrogate-based explanations for individual predictions based on sampling centered on a particular relevant place of the decision boundary, rather than on the prediction itself. This allows them to achieve substantially better results, demonstrated visually on the UCI half moons datasets, where the local explanation often does not agree with the global explanation due to an unusually shaped decision boundary. Garreau et al. [3] derive closed-form expressions for linear explanation systems and report that the coefficients are proportional to the gradient of the function being explained. Reiger et al. [13] present evidence that aggregate explanations are more robust to attacks than individual explanation methods. We use existing work [3, 7, 11, 12] along with concepts from linear algebra to propose a concise, informative new format for global explanation representation which can be employed in places where a condensed version of how model logic varies across the region of interest is a requirement.

3 EQUI-EXPLANATION MAPS

Let f be a black-box binary classifier that maps input features $\mathcal{S} = \{x_1, x_2, \ldots, x_n\}$ to $\{0, 1\}$. The input features can be either tabular or categorical. Following other explanation models [12], the domain of an explanation model is the subset of interpretable features $\mathcal{E}$ where $\mathcal{E} \subseteq \mathcal{S}$ and $|\mathcal{E}| = d$. Assume that we have knowledge of a finite region of interest of $\mathcal{E}$, named P. The region of interest for explanation purposes may be smaller than the actual range of feature values. For example, healthcare providers would want to study heart disease symptoms in patients with age ranging from 0 to 120, instead of ages ranging from $-\infty$ to $+\infty$.

We globally explain f by providing a division of region P into ϵ-equi-explanation subspaces. An ϵ-equi-explanation subspace is

[1] https://github.com/TaKneeAa/EquiExplanationMaps

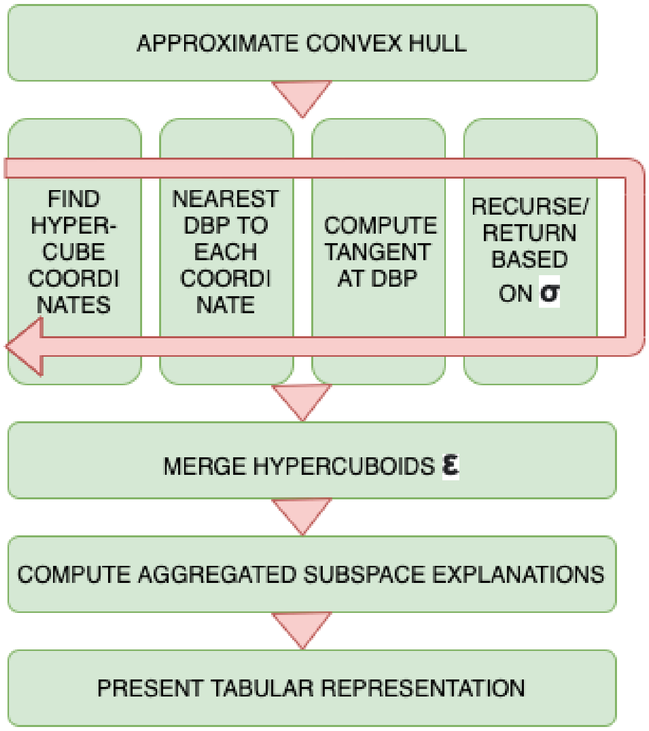
APPROXIMATE CONVEX HULL
FIND HYPER-CUBE COORDI NATES
NEAREST DBP TO EACH COORDI NATE
COMPUTE TANGENT AT DBP
RECURSE/ RETURN BASED ON σ
MERGE HYPERCUBOIDS ε
COMPUTE AGGREGATED SUBSPACE EXPLANATIONS
PRESENT TABULAR REPRESENTATION

Figure 2: E-Map architecture to generate ϵ-equi-explanation maps. Given a binary classifier f as a black-box and region of interest P, we try to partition $\mathcal{E}$ into equi-explanation regions in accordance to the decision boundary of f. We compute the approximate convex hull of $\mathcal{E}$ and run a divide-and-conquer based approach, where in each recursion step we compute the explanation vectors of hyper-cuboid coordinates. Next, based on the standard deviation of the hyper-cuboid vertices exceeding ϵ we decide if we want to divide the hyper-cuboid into sub hyper-cuboids. After the recursion ends, we try to merge hypercuboids, if their standard deviation is less than ϵ. We lastly do a weighted aggregate of hyper-cuboid explanations to assign an explanation for a subspace.

defined as a subspace of the explanation space where the deviations of local explanations for points within that subspace, do not exceed ϵ. Specifically, classifier f is explained as $[\mathbf{e}, \mathbf{W}]$ where $\mathbf{e}$ is a partition of the explanation feature hyperspace $\mathcal{E}$ into ϵ-equi-explanation subspaces and each subspace $e_i \in \mathbf{e}$ is explained with a function $\mathbf{w}_i \in \mathbf{W}$. Each explanation function $\mathbf{w}$ belongs to a class of potentially interpretable models, for which we consider linear functions in this study. Thus, the explanation for each subspace $\mathbf{w}_i \in \mathbb{R}^d$ represents the contribution of each explanation feature towards the model decision within that subspace. Standard deviation of linear explanation vectors is measured to check whether a subspace is ϵ-equi-explanation or not.

To obtain ϵ-equi-explanation subspaces, we propose a divide-and-conquer approach, **E-Map**, summarized by the pseudo-code given in Algorithm 1. E-Map computes the hyper-cuboid of desired ranges of explanation features P and divides it into sub-hypercuboids if it is not an ϵ-equi-explanation space. The obtained sub-hypercuboids are then recursively checked and divided if necessary. When all the obtained hypercuboids are ϵ-equi-explanation subspaces, E-Map checks if neighboring subspaces can be merged to reduce the number of partitions. Each explanation subspace e thus consists of a set of hyper-cuboids in the d-dimensional space, and is linearly explained by a weighted average of each constituent hyper-cuboid's explanation vector. We describe the E-Map approach in detail below.

Algorithm 1 Pseudo-code of E-Map approach to generate ϵ-equi-explanation maps

```
Require: $f$: binary classifier, $\mathcal{E} \in \mathbb{R}^d$: explanation features, $P$: Region of interest for explanation, $\epsilon \in [0,1]$
C = ConvexHull($P$)
CPoints = Vertices($C$)
procedure Divide-hyper-cuboid(CPoints)
    DBP = DecisionBoundaryPoints(CPoints)
    ExplanationVectors = DBPTangents(DBP)
    $\sigma$ = FindDeviation(ExplanationVectors)
    if $\sigma > \epsilon$ then
        list = CreateSubCuboids(CPoints)
        for CPoint in list do
            CPoints,ExplanationVectors = Divide-hyper-cuboid(CPoint)
    return CPoints, ExplanationVectors
CPoints,ExplanationVectors = Divide-hyper-cuboid(CPoints)
CPoints,ExplanationVectors = Merge-Cuboids(CPoints,ExplanationVectors)
e,W = Aggregated-Subspace-Explanation(Cpoints,ExplanationVectors)
```

Convex hull of region of interest. The first step of E-Map is to compute the convex hull of our d-dimensional region of interest P. Finding an exact convex hull is an NP-hard problem, however, an approximate convex hull of the region is sufficient for our purpose. As a result, we compute the hull coordinates, using the maximum and minimum values of each feature in the region of interest, and get a hyper-cuboid hull of 2^d vertices.

Divide-and-conquer algorithm. Starting with the hyper-cuboid hull of the region of interest, we use a divide-and-conquer algorithm to obtain an ϵ-equi-explanation map. Given a hyper-cuboid hull, the function `divide-hyper-cuboid` in Algorithm 1 computes a local explanation of classifier f at each vertex of the hull. It then determines if the explanation vectors are similar enough to label the hypercuboid as $\epsilon-$equi-explanation. If the explanation vectors are similar enough, the hyper-cuboid is an equi-explanation subspace. Otherwise, the hyper-cuboid is partitioned into two sub-hyper-cuboids, splitting along the hyper-cuboid plain with the most explanation distance between its faces in the middle. Then, the function `divide-hyper-cuboid` is called recursively for each of the partitions.

Generating local explanations. Approaches like LIME use a falling exponential kernel defined on a distance metric to weigh the importance of each perturbed input as in Equation (3). On examining Equations (2) and (3), we observe that the decision boundary

Algorithm 2 Growing spheres algorithm for computing Decision Boundary Point nearest to x [7]

```
Require: $f : \mathbf{x} \rightarrow \{0, 1\}$: a binary classifier , $x \in \mathbf{x}$: an observation to be explained, $\eta, n$: Hyperparameters
Ensure: Nearest Decision Boundary Point $e$
Generate $(z_i)_{i \leq n}$ uniformly in $\mathrm{SL}(x, 0, \eta)$
while $\exists k \in (z_i)_{i \leq n} : f(k) \neq f(x)$ do
    $\eta = \eta/2$
    Generate $(z_i)_{i \leq n}$ uniformly in $\mathrm{SL}(x, 0, \eta)$
$a_0 = \eta, a_1 = 2\eta$
Generate $(z_i)_{i \leq n}$ uniformly in $\mathrm{SL}(x, a_0, a_1)$
while not $\exists k \in (z_i)_{i \leq n} : f(k) \neq f(x)$ do
    $a_0 = a_1$
    $a_1 = a_1 + \eta$
    Generate $(z_i)_{i \leq n}$ uniformly in $SL(x, a_0, a_1)$
return $k$, the $l_2$-closest decision boundary point from $x$
```

point (DBP) nearest to the instance being explained plays the most significant role in generating explanations. Points further than the nearest DBP will have little influence due to the rapidly falling negative exponential function. However, as Laugel et al. [7] point out, the solution by LIME largely depends on the density of sampling and the proximity being sampled. Considering the scale of our task, a method that generates approximate explanations but with less variability, would yield better results. Therefore, we compute the tangent to the DBP nearest to the instance to be explained for obtaining the local linear explanation. This method leads to a solution approximate to what is returned by LIME, if the instance neighborhood has been thoroughly sampled by LIME. However, with a higher number of dimensions dense sampling around an instance is hardly practical. The proposed approach is thus expected to return solutions better than LIME with poor sampling.

Nearest DBP_x. To find the point nearest to an instance x on the decision boundary of classifier f, we tweak the Growing Spheres algorithm proposed by Laugel et al. [7]. We uniformly sample points in the sphere centered at x with the radius of η, which is initially set to a large value. In order to sample points inside $\mathrm{SL}(x, a_0, a_1)$, we sample observations uniformly distributed over the surface of a unit sphere, then draw $\mathcal{U}(a_0, a_1)$-distributed values and use them to re-scale the distances between the sampled observations and x. In order to sample uniformly on a unit sphere, we sample observations from $\mathcal{N}(x, 1)$ and scale them to a distance of 1 from x. Any of the sampled points having a class different than x guarantees the presence of at least one DBP within the sphere. However, this may not be the nearest DBP to x. As a result, the process is repeated by sampling points in a spheres of radius $\eta = \eta/2$ and keep halving η until we have a sphere where all sampled points have the same class label as x. This indicates the absence of a decision boundary in the sphere. Next we sample points inside the spherical layer $\mathrm{SL}(x, \eta, 2\eta)$ in search of points which have a class label different from x. Out of the sampled points which have a different label, we accept the one with the minimum $L2$-distance to x as the nearest DBP. If no such point is found within this spherical layer, then we search in the next spherical layer $\mathrm{SL}(x, 2\eta, 3\eta)$, and so on until the nearest DBP is found. Let us name this point DBP_x. The pseudo-code for this process by [7] is provided in Algorithm 2.

Tangent at DBP_x. Finding DBP_x, we then compute the tangent of the decision boundary of f at this point. To compute this, we randomly perturb point DBP_x and feed the perturbed instances to f to get their predicted labels. A weighted linear regression model is then learned on perturbed instances to obtain the tangent of f at DBP_x. We use the kernel function from KernelSHAP[11] as the distance metrics in this regression (Equation 4) along with the Loss function defined in Equation 2.

$$\pi_x(z) = \frac{d-1}{({}^dC_{|z|})|z|(d-|z|)}, \tag{4}$$

where $|z|$ is the number of non-zero elements in the d dimensional vector z. This results in a d-dimensional tangent that locally explains instance x.

ϵ**-equi explanation subspaces**. We define the standard deviation σ_{EV} for a set EV of explanation vectors (Equation 5) as:

$$\sigma_{EV} = \sum_{\forall p_i, p_j \in EV} \sum_{m \in d} (p_{im} - p_{jm})^2. \tag{5}$$

We next compute σ_{EV}, where EV stands for the set of explanation vectors of a hyper-cuboid's coordinates. If σ_{EV} exceeds the value of the hyperparameter ϵ, the obtained region cannot be termed an ϵ-equi-explanation subspace and is further divided. For further division, we iterate over each explanation feature ($i \in d$) and find the average explanation vector corresponding to its minimum and maximum values ($X_{i_{min}}$ and $X_{i_{max}}$). For every feature $i \in d$, we then compute the $L2$ distance between $X_{i_{min}}$ and $X_{i_{max}}$. We pick the feature (f) with the highest distance (Equation 6), and partition the hyper-cuboid into two across that feature.

$$f = \arg\max_i (X_{i_{min}} - X_{i_{max}})^2 \tag{6}$$

If σ does not exceed ϵ, an $\epsilon-$equi-explanation subspace has been found. When the recursion ends, a set of hyper-cuboids are obtained where the standard deviation of each hyper-cuboid is less than ϵ. At this step, a merge function is used to check if any two hyper-cuboids can be merged while still satisfying the ϵ constraint. These merged hypercuboids represent subspaces $\mathbf{e}$ of equi-explanation maps.

Subspace linear explanation. Once we have obtained ϵ-equi-explanation subspaces, each being a set of hyper-cuboids, we generate a linear explanation for each subspace depicting the behavior of f in that subspace. For this step, we first compute an explanation vector for each hyper-cuboid by computing the explanation vector for each of the hyper-cuboid's coordinates and averaging them. We additionally compute the volume of each hyper-cuboid (product of edge lengths). To generate an aggregated explanation for each subspace, we average each hyper-cuboid's explanation vector weighted by its volume. This leads to a division of the input space into ϵ-equi-explanation maps with normalized explanation vectors for each subspace.

Presentation of results. E-Map partitions the space into subspaces where each subspace is a union of d-dimensional hyper-cuboids with d being the number of explanation features. For models with fewer than 3 explanation features, equi-explanation maps can be visualized as in Figure 1. However for models with greater

Table 1: A representation of equi-explanation maps for a XGBoost classifier trained on the UCI Heart disease dataset, using three explanation features: Age, RestECG, and Cholesterol. We depict the four explanation regions (subspaces) generated by our algorithm and present the approximate model logic corresponding to each of the four regions.

Subspace		Age	Cholesterol	RestECG	Explanation
1	Min	29	285	0	[0.26,**0.68**,0.06]
	Max	44	364	2	
2	Min	44	126	0	[0.21,0.25,**0.54**]
	Max	63	285	0	
	Min	29	364	1	
	Max	44	564	2	
	Min	63	284	2	
	Max	77	364	2	
3	Min	63	126	0	[**0.45**,0.2,0.35]
	Max	77	284	1	
4	Min	29	126	0	[0.33,0.31,**0.36**]
	Max	44	285	1	
	Min	44	285	1	
	Max	63	364	2	
	Min	63	364	2	
	Max	77	564	2	

Table 2: Accuracy of the four chosen classifiers on the training and test sets of the Heart disease and Pima Indians datasets.

Dataset	Algorithm	Training Accuracy	Test Accuracy
Heart Disease	Logistic Regression	0.86	0.80
	SVM	0.92	0.80
	MLP	1	0.81
	XGBoost Classifier	1	0.78
Pima Diabetes	Logistic Regression	0.79	0.73
	SVM	0.83	0.72
	MLP	1	0.77
	XGBoost Classifier	1	0.75

than 3 explanation features, we present the coordinates of each hyper-cuboid in a compact tabular representation using $2d$ numbers for each hyper-cuboid, to enable at-a-glance summaries (as shown in Table 1). The value of ϵ can be set according to the granularity of explanations required.

4 EXPERIMENTS

To the best of our knowledge, our work is the first towards generating explanations that summarize the model logic of a black-box classifier over different regions of feature values. However, since it lies in the large space of global explanation methods, we compare the quality of E-Map generated equi-explanation maps with other global explanation methods.

4.1 Baseline Models

SP-LIME [12], an extension of LIME, is a model-agnostic approach that chooses diverse and representative instances to describe the global model logic. Given a budget B, SP-LIME selects $|B|$ instances from a uniformly sampled set X using a greedy approach based on the local explanation for each instance.

Guided-LIME [15] adds a structured-sampling preprocessing step to the input of SP-LIME in order to improve the fidelity of LIME-based approaches. To do this, they employ Formal Concept Analysis (FCA) assuming access to the complete model training data. Generating SP-LIME explanations on the full dataset (especially for tabular features) has large computational complexity, which Guided-LIME successfully reduces.

SHAP [11], originally a local feature attribution method, is extended for global explanation of tree-based models [10]. The authors report that it outperforms existing explainers on various metrics like run time, accuracy, consistency guarantees, mask, resample, and impute for tree based models.

MUSE [6] is a rule-based mimic model [1] explanation algorithm to explain how a model behaves in subspaces characterized by certain features of interest. It aims to learn compact *decision sets*, each of which is a series of if-then rules built by optimizing for fidelity, unambiguity, and interpretability. Since MUSE and equi-explanation maps have representational differences, we compare equi-explanation maps to MUSE only by the user study. We do not include comparison with interpretable decision sets (IDS) [5] and Bayesian decision lists (BDL) [9], as they have been shown to under-perform MUSE [6].

4.2 Evaluation Metrics

To enable comparison of equi-explanation maps to other linear global explanation algorithms, each with a different representation format, we propose the following general evaluation metrics.

Interpretability: The multiplicative inverse of the amount of information (numbers) needed to present generated explanations to users. It is dependent on the format of explanation presentation by an explanation algorithm. For example, if a model has 4 explanation features and is to be explained with 3 representative instances, its Emap interpretability would be $\frac{1}{3*(2*4+4)}$ whereas its LIME interpretability would be $\frac{1}{(3*4)}$. The higher the score, the more interpretable the explanation algorithm.

Fidelity: The fraction of sampled points from the region of interest for which the black-box prediction agrees with the *explanation model prediction*. For explanation models that generate representative instances, the reconstructed *explanation model prediction* is either the prediction by the explanation vector of its subspace (if subspaces are defined) or the prediction by the explanation vector of its nearest representative element. The higher the fidelity, the better the explanation algorithm.

Informativeness: The average similarity between local and subspace explanation vectors for points uniformly sampled in the region of interest. The local explanation of a sampled instance is computed using LIME. The subspace explanation is either the explanation vector of the subspace (if subspaces are defined) or the explanation vector of the representative unit nearest to the instance. Similarity is computed using cosine similarity. Higher this metric, more informative the summary.

4.3 Experimental Settings

Datasets : We perform our experiments on two real-world datasets from the medical domain. The motivation is the known importance of subspace explanations in clinical diagnostic settings [5]. The first dataset is the *UCI Heart Disease dataset*[2], which has 303 instances and 14 real valued features designed to predict the presence (labels: 1,2,3,4) and absence (label: 0) of heart disease.

Secondly we use the *Pima Indians Diabetes* dataset[3], predicting the onset of diabetes within 5 years in Pima Indians, given their medical details. It is a binary classification dataset with 768 observations containing 8 input features and a binary output. The label for each instance is either 0 or 1, with 1 indicating that the person would see an onset of diabetes within 5 years.

Classifiers : We train binary classifier models on the Heart Disease and Pima Diabetes datasets using four algorithms Logistic Regression, SVM, MLP, and XGBoost. We use the scikit-learn implementation of logistic regression. We learn a Support Vector Machine with an RBF kernel using the scikit-learn implementation. We experiment with different configurations of MLP using PyTorch. We settle on an architecture with 3 hidden layers. For a dataset of N dimensions, the MLP has N neurons in the first layers, $2 * N$ in the second, N in the third, $\frac{N}{2}$ in the fourth and a single neuron in the final output layer. For the XGBoost classifier, we use `XGBClassifier` from `xgboost`. The prediction threshold is set to 0.5, i.e., the prediction is 1 if scores are greater or equal to 0.5, otherwise the prediction is zero[20]. We split both datasets into three parts: train, validation, and test in the ratio of 80:10:10. We use five-fold cross validation on the training data to learn the supervised models. We compute the classification accuracy of each of these algorithms and report results in Table 2.

Setting 1 : In the first set of experiments, we compare the performance of Equi-explanation maps with respect to other global linear explanation methods on the two chosen datasets. We carry out experiments with all input features as explanation features and with a budget of four representative instances (as in SP-LIME). In order to carry out fair comparison, we tune E-Map with different values of ϵ until four subspaces are generated. For the sake of uniformity, we also demarcate the centroid of each subspace in E-Map as its representative vector. SP-LIME and Guided-LIME algorithms output representative instances and their respective explanations as output, but no subspace information. Equi-explanation maps output representative instances, their explanations, and the coordinates of subspace hyper-cuboids (refer to Table 1). Both SP-LIME and Guided-LIME require a sampling density to determine the number of perturbations, the value of which is retained from the original LIME repository. For E-Map, we set initial sampling radius η to 1 and the number of points to sample on the sphere η to $1,000$ (following recommendations by Laugel et al. [7]). For the comparing methods, we set as many parameters as possible to the values reported in their original drafts or repositories. MUSE is a mimic model-based explanation method whose performance is difficult to compare with representation vector-based explanation empirically. As a result, we put off comparison of E-Map with MUSE to a user study (Section 5). For our evaluation metrics *Fidelity* and *Informativeness*, we uniformly sample 500 points from the region of interest. The same 500 points are used to evaluate all competing systems for a given dataset and classifier.

Setting 2 : For the next set of experiments, we compare the equi-explanation maps generated by E-Map for the four different classifiers described above. Setting the value of ϵ to a fixed 0.6, the aim of this experiment is to study the variation in *Interpretability*, *Fidelity*, and *Informativeness* for explanations of classifiers with different complexities on the two chosen datasets. As above, we set initial sampling radius η to 1 and the number of points to sample on the sphere η to $1,000$ (as above). We again sample 500 points from the region of interest and use the same points to evaluate explanations for different classifiers.

4.4 Results and Observations

The results of our experiments comparing equi-explanation maps generated by E-Map with representation-vector based explanations by other global explanation methods are reported in Table 3. The representative instances returned by equi-explanation maps show 43% and 38% higher fidelity than those returned by SP-LIME and Guided-LIME, respectively. This indicates that a user is more likely to guess the black-box model's decision for an instance, when shown an E-Map explanation as compared to when shown a *-LIME explanation.

This might be attributed to equi-explanation maps presenting subspaces of complex shapes and sizes considering intricacies of decision boundaries compared to the spherical subspaces carved by *-LIME.

E-Map explanations are also 41% and 21.5% more informative as compared to SP-LIME and Guided-LIME explanations, respectively. This indicates that a user is more likely to accurately guess the explanation of an unseen instance when shown an equi-explanation map explanation rather than the when shown other kinds of explanations. This might again be attributed to complex subspace boundaries for equi-explanation regions as compared to other explanation algorithms.

Due to the extra reporting of subspace coordinates, equi-explanation maps show 79.6% lesser interpretability compared to the other approaches. Since SP-LIME and Guided-LIME only return representative instances and their explanations, their interpretability score for a budget B is computed as $|B|$*(NUMBER OF INPUT FEATURES + NUMBER OF EXPLANATION FEATURES). The interpretability of E-Map, which returns hyper-cuboid dimensions, additionally includes an extra $|B|$*(2*NUMBER OF EXPLANATION FEATURES*NUMBER OF HYPER-CUBOIDS) term. As a result, although we see a gain in Fidelity and Informativeness with E-Map, we see a drop in Interpretability.

The results of comparing E-Map explanations for different classifiers is presented in Table 4. Over experiments with E-Map on different classifiers, we observe a strong correlation between the number of subspaces returned by E-Map and the complexity of the classifier being explained, for a fixed value of ϵ. Overall the equi-explanation maps' explanation of the simplest classifier (logistic regression) achieves 105% more interpretability, 79% more fidelity, and 40% more informativeness as compared to that of the classifier

[2] https://archive.ics.uci.edu/ml/datasets/Heart+Disease
[3] https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.csv

Table 3: Comparing E-Map generated equi-explanation maps to existing global explanation algorithms with a budget of 4 representative instances on all explanation features for two medical domain datasets, averaging performance over all four classifiers.

	UCI Heart Disease			PIMA Indian Diabetes		
Method	Interpretability	Fidelity	Informativeness	Interpretability	Fidelity	Informativeness
SP-LIME	0.083	0.49	0.56	0.083	0.56	0.63
Guided-LIME	0.083	0.62	0.65	0.083	0.62	0.66
SHAP	0.083	0.56	0.66	0.083	0.60	0.69
E-Map	0.017	**0.86**	**0.79**	0.017	**0.88**	**0.86**

Table 4: Comparing E-Map generated equi-explanation maps to explain four different classifiers on two medical domain datasets for a fixed value of ϵ set to 0.6.

	UCI Heart Disease			PIMA Indian Diabetes		
Method	Interpretability	Fidelity	Informativeness	Interpretability	Fidelity	Informativeness
Logistic Regression	0.037	0.97	0.91	0.050	0.99	0.95
SVM	0.018	0.56	0.65	0.025	0.69	0.82
3-layer MLP	0.032	0.82	0.82	0.040	0.87	0.93
XGBoost Classifier	0.021	0.80	0.79	0.028	0.93	0.87

with the most complex decision boundary (here: SVM) for $\epsilon = 0.6$. This is intuitive as the more curved the decision boundary is, the greater the deviation in explanations is and vice versa.

MUSE is a series of cascading if-then rules, with a maximum hierarchy of two layers. For datasets with mostly tabular features, MUSE will have exponential combinations over feature ranges and would be pretty unintuitive to observe. Decision set-based approaches seem to work well for datasets with mostly categorical features, like the datasets chosen in demonstrated examples of previous studies [5, 6, 9]. Since many real world datasets are largely tabular, we believe equi-explanation maps would provide better explanations for them as compared to decision sets-based approaches [5, 6, 9].

5 USER STUDY

In the previous section, we demonstrated that our approach outperforms the compared baselines on grounds of Fidelity and Informativeness. However different global explanation algorithms result in different formats, making it hard to compare algorithm performances. In order to enable fair comparison between rule-based mimic model explanations (MUSE) and equi-explanation maps, we conduct a user study. [4] Inspired by explainable AI literature, we recruited 10 students who had completed at least one undergraduate level machine learning course. Two different XGBoost classifiers were trained on non-overlapping sections of the UCI heart disease dataset and the UCI Divorce predictors dataset and used to generate two explanation presentations: an equi-explanation map and a falling if-then list. The volunteers were then asked a series of questions based on both explanation presentations.

On the basis of each explanation presentation, the volunteers were asked to: (i) predict the classifier label for a given instance (ii) given a partition of the feature space (e.g. age > 30), indicate the primary features used in decision making for that partition. The ground truth for (i) was computed by feeding the instance to the black-box classifier. The ground truth for (ii) was obtained by generating local LIME explanations for instances in that partition. The user's response accuracy to these questions was then computed. We observe that MUSE achieves a slightly higher accuracy for question 1, outperforming equi-explanation maps by 8.5%. However, the equi-explanation map group outperforms the MUSE group by 46% on accuracy in question 2.

Participants were also asked (i) Which presentation do you think would give more compact explanations as the number of feature scales? (ii) Which explanation presentation did you find more informative? (iii) Which explanation presentation did you find easier to understand? 100% volunteers answered Equi-explanation maps for (i), 70% for (ii), and 30% for (iii).

This shows that if a user wants to understand what features influence the decision making in a certain region (which is the primary intent behind generating explanations), equi-explanation maps should be unarguably preferred.

6 DISCUSSION

Local explanations are useful to understand the model behavior for a specific instance, but do not tell much about the larger picture. Most existing global explanation methods on the other hand are too sparse and not very informative about the variation in model logic across the region of interest [7]. Global explanations that summarize the model logic using equi-explanation maps lie somewhere in between, with an informativeness-interpretability trade-off, which can be set by tuning ϵ as per user requirements.

Most existing explanation work focuses solely on *Interpretability* and *Fidelity/Faithfulness* as the primary metrics to optimize during explanation generation. However, we believe that *Informativeness* - which is a proxy for the knowledge gained is an important metric to consider as well. While the *Fidelity* metric is an evaluation metric in the black-box decision space, *Informativeness* is an evaluation metric in the explanation space. We believe that lower bounds and

[4] UMass Amherst IRB (IRB Number 3410).

confidence intervals on Fidelity and Informativeness should be mandated in instances where the explanation is highly consequential: e.g. deciding between treatment options by a doctor, deciding jail term length. We believe that using summaries of the black-box logic for global explanations instead of existing approaches would induce more trust in users in the above mentioned scenarios.

Due to the increased number of bits in Equi-explanation maps explanations, they are more suitable in low dimensional settings. However, they can still be generated in higher dimensional settings to verify black-box model behavior. For example, a developer would prefer an informative summary explanation to verify how their model prioritizes features on all subspaces, even at the cost of interpretability.

Studying explanations with a subset of input features as explanation features might not always be as insightful even if it is more interpretable. For example, for a classifier with input features [A,B,C] if we generate explanations using only feature A and B, there might be a causal feature C driving model decision making for that variable. However, these factors depend on the exact problem statement in hand.

7 RELATED WORK

Recently Setzu et al. [18] propose GLocalX, a tool that adds an interpretable layer on top of a black-box by aggregating local explanations agnostic to the model being explained. GLocalX hierarchically aggregates the local explanations, represented as decision rules with the goal of emulating the black-box. Their output format is similar to MuSe [6] and serves as a mimic model to the black-box explanation. Our proposed data structure, subspaces as a union of hyper-cuboids can also be visualized using Polyhedral Sets from linear algebra. Ruggieri et al. [14] propose a method of learning a parameterized linear system whose class of polyhedra includes a given set of example polyhedral sets and it is minimal.

Apart from explanations for black-box models, certain algorithms generate explanations while taking the model architecture into consideration. One such notable model-introspective explanation method for deep learning models is DeepLIFT [19] which backpropagates the output of DNNs to assign each input feature a contribution weight. DeepLIFT [19] specializes in that its assigned feature weight, positive or negative, can be computed in a single backward pass of the neural network. Selvaraju et al. [17] introduce Grad-CAM, a method for generating *visual explanations* across the layers of a convolutional neural network (CNN), using target gradients to create coarse localization maps highlighting region importance. SHAP [11] provides unifying framework and formalizes explainability using the Shapely values principle from game theory. The SHAP method assigns each feature an importance value for a particular prediction. It is notable because it proves that there exists a unique solution in this class with a set of desirable properties. Their framework unifies six existing methods including LIME [12] and DeepLift [19] discussed above. Apart from the local explanation techniques discussed so far, there are a few algorithms that help users make global conclusions about the model.

The above methods focus on features which are present, even though these features might have a negative contribution in the classification. A few recent works have been focused on identifying features which are necessary or sufficient to explain an instances classification by a model. Dhurandhar et al. [2] present contrastive explanations to explain black-box classification models. Given an input, they find what features must be *minimally and sufficiently present* and *minimally and sufficiently absent* to justify its classification. The authors argue that this format of explanations is more in line with the human way of thinking. Another recent branch of model agnostic black-box explanation methods include counterfactual explanations. Given a *query* image I, for which a vision system predicts class C, a counterfactual visual explanation identifies how I could change such that the system would output a different specified class [4]. Looveren et al. [21] further propose a method to speed up the the search for counterfactual instances to generate interpretable counterfactual explanations. All the above efforts discuss post-hoc interpretability methods. Recently a lot of interest has arised in generating causal explanation algorithms, especially with applications in genomics [22]. Algorithms like CXPlain [16] use a causal learning function to train the surrogate model and combine it with bootstrap to measure uncertainty in explanations.

8 CONCLUSION

In this work, we proposed the new paradigm of summarizing the model logic of a black-box in order to generate global explanations. In order to do this, we propose Equi-explanation maps, a novel concise representation for global explanations. We further proposed E-Map, an effective method that generates equi-explanation maps. Using hyper-cuboids as units of equi-explainability, we termed the union of hyper-cuboids to be a subspace and assigned a linear explanation to each subspace. We experimented on two medical records datasets: the UCI heart disease dataset and the Pima Indians Diabetes dataset and our approach was evaluated using the metrics interpretability, fidelity and informativeness, and substantially outperformed competitive methods in most of these. With this work, we introduce the task of *Global summary explanation generation* - explanations which present a summary of the model logic of a black-box model. We hope future explainability researchers study the compactness-informativeness trade-off for global summaries and propose better ways to generate summary explanations. It would also be more effective to see equi-explanation maps being generated using causal explanation techniques with bounds on subspace uncertainty.

ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF grant IIS-2039449 and in part by NSF grant number 1813662. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Nadia Burkart and Marco F Huber. 2021. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* 70 (2021), 245–317.

[2] Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. 2018. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. *arXiv preprint arXiv:1802.07623* (2018).

[3] Damien Garreau and Ulrike Luxburg. 2020. Explaining the explainer: A first theoretical analysis of LIME. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1287–1296.

[4] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Counterfactual visual explanations. In *ICML*. PMLR, 2376–2384.

[5] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. 2016. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1675–1684.

[6] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. 2019. Faithful and customizable explanations of black box models. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 131–138.

[7] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. 2018. Comparison-based inverse classification for interpretability in machine learning. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer, 100–111.

[8] Thibault Laugel, Xavier Renard, Marie-Jeanne Lesot, Christophe Marsala, and Marcin Detyniecki. 2018. Defining locality for surrogates in post-hoc interpretablity. *ICML’18 Workshop on Health* (2018).

[9] Benjamin Letham, Cynthia Rudin, Tyler H McCormick, David Madigan, et al. 2015. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics* 9, 3 (2015), 1350–1371.

[10] Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. 2020. From local explanations to global understanding with explainable AI for trees. *Nature machine intelligence* 2, 1 (2020), 2522–5839.

[11] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*. 4765–4774.

[12] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 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.

[13] Laura Rieger and Lars Kai Hansen. 2019. Aggregating explanation methods for stable and robust explainability. *arXiv preprint arXiv:1903.00519* (2019).

[14] Salvatore Ruggieri. 2013. Learning from polyhedral sets. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

[15] Amit Sangroya, Mouli Rastogi, C Anantaram, and Lovekesh Vig. [n. d.]. Guided-LIME: Structured Sampling based Hybrid Approach towards Explaining Blackbox Machine Learning Models. ([n. d.]).

[16] Patrick Schwab and Walter Karlen. 2019. Cxplain: Causal explanations for model interpretation under uncertainty. *arXiv preprint arXiv:1910.12336* (2019).

[17] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.

[18] Mattia Setzu, Riccardo Guidotti, Anna Monreale, Franco Turini, Dino Pedreschi, and Fosca Giannotti. 2021. GLocalX-From Local to Global Explanations of Black Box AI Models. *Artificial Intelligence* 294 (2021), 103457.

[19] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features through Propagating Activation Differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (Sydney, NSW, Australia) *(ICML’17)*. JMLR.org, 3145–3153.

[20] ShubhankarRawat. 2016. ShubhankarRawat/Heart-Disease-Prediction. https://github.com/ShubhankarRawat/Heart-Disease-Prediction. (2016). [Online; accessed 11-June-2021].

[21] Arnaud Van Looveren and Janis Klaise. 2019. Interpretable counterfactual explanations guided by prototypes. *arXiv preprint arXiv:1907.02584* (2019).

[22] David Watson. 2021. Interpretable Machine Learning for Genomics. (2021).