Interactive Visual Self-service Data Classification Approach to Democratize Machine Learning

Sridevi Narayana Wagle
Dept. of Computer Science
Central Washington University
Ellensburg, Washington, USA
Sridevi.Wagle@cwu.edu

Boris Kovalerchuk
Dept. of Computer Science
Central Washington University
Ellensburg, Washington, USA
BorisK@cwu.edu

***Abstract*—Although machine learning algorithms are progressively used in an expansive range of domains, the effective machine learning classifiers are often black-boxed, non-comprehensive to the end users and beyond their abilities to develop models themselves. To overcome this challenge, data visualization combined with self-service or democratized machine learning is proposed in the form of the Iterative Logical Classifier (ILC) algorithm with an added advantage of outperforming the accuracies of black-box machine learning classifiers on benchmark datasets. The algorithm is based on the concept of Shifted Paired Coordinates that allow 2-D visualization of n-D data without loss of n-D information.**

Keywords—Self-service Machine Learning, Interactive Data Visualization, Logical Classifier, AutoML.

I. INTRODUCTION

Democratization of Machine Learning (ML) with automated machine learning (AutoML) progressed significantly in optimizing deep learning models to be used by *model developers* who are not ML experts, e.g., [4]. The wider goal is the accessibility of democratized machine learning to the *domain experts* to analyze the data better [5] so that they can classify the data with more confidence. Black-box models do *not explain* them in their *domain terms* how the classification output is obtained, which is critical in the medical domain, for instance, while dealing with sensitive cancer data. In such cases it is difficult to rely on a black-box classifier to make an informed decision. This paper describes an efficient approach to provide the end user with useful interactive visualization of data to aid analysis. With the help of visualization, domain experts can get a good understanding of how the multidimensional n-D data are distributed and can identify the patterns in the graph visually where good separation of classes are observed. The end users can then decide on different criteria or rules to classify the data using *Iterative Logical Classifier* (ILC) algorithm.

II. INTERACTIVE SHIFTED PAIRED COORDIANTE SYSTEM

Multidimensional n-D data can be represented *losslessly* using new Shifted Paired Coordinates (SPC) [6], which show the n-D data by a graph in the sequence of shifted pairs of dimensions in the two-dimensional plane. The *Interactive Shifted Paired Coordinate software system* (SPCVis) allows the user to visualize the data in SPC. In this paper, the data separation is primarily illustrated on the vertical axes to simplify the attention of the user. A user can assign specific axes as vertical ones and then look for a good separation of the data. In general terms, it requires several trials to set the vertical axes.

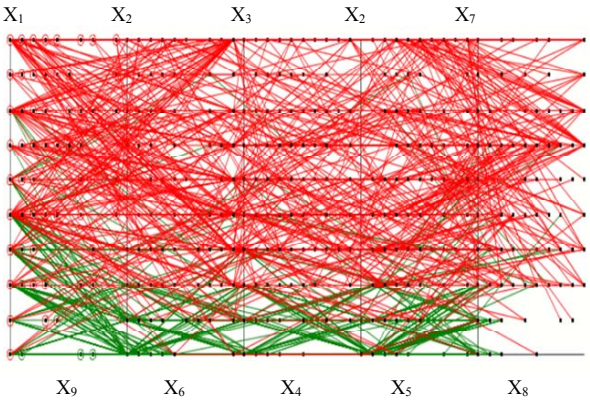

Fig. 1: WBC 9-D dataset visualized in SPC.

Fig. 1 shows 9-D Wisconsin Breast Cancer (WBC) dataset [2] in SPC with such selection of vertical axes. Green lines represent class 1 (benign) and red lines represent class 2 (malignant). The user is also provided with an interactive feature called *non-linear scaling* where only a part of the user selected coordinate is scaled differently. The generalized formula for an n-D point x_3 is as given below in equation (1), where k is a constant and $0 < k < 1$. The value of k is set by the user. The data used for SPC are normalized to [0, 1].

$$x_3' = \begin{cases} x_3, & \text{if } x_3 < k \\ x_3 + 0.1 \times graphWidth, & \text{if } k \le x_3 < 1 \end{cases} \quad (1)$$

Our experiments with several datasets and SPC show that such non-linear scaling allows improving visual discrimination of classes. Using interactive visualization alone does not completely perform the data separation. It only provides a base for the separation by providing threshold values for class separation. Using these threshold values, data separation can be performed more completely using the *Iterative Logical Classifier* algorithm by creating a set of analytical rules based on these threshold values.

III. ITERATIVE LOGICAL CLASSIFIER ALGORITHM

Iterative Logical Classifier (ILC) is an algorithm that performs data separation in iterations. In every iteration, the

978-1-7281-9134-8/20/$31.00 ©2020 IEEE
DOI 10.1109/IV51561.2020.00052

data are redisplayed and reorganized to find good class separation patterns. The steps performed for the classifier are:

Step 1: Compute threshold values based on data distribution using the interactive controls in SPC software (switching coordinates, non-linear scaling etc.)

Step 2: Create analytical rules like in [7] interactively or automatically for the data of individual classes. For an n-D point $\mathbf{x} = (x_1, x_2 x_3, \ldots, x_n)$ to be classified to a class:

$$\text{If } x_i \in R_{class1}, \text{ then } \mathbf{x} \in \text{class 1} \quad (2)$$

$$\text{If } x_j \in R_{class2}, \text{ then } \mathbf{x} \in \text{class 2} \quad (3)$$

where R_{class1} and R_{class2} are a set of SPC areas generated using the threshold values.

Step 3: For the n-D data points **x** that do not follow equations (2) and (3), separation is performed in the next iteration.

Step 4: Step 1 – 3 is repeated on the remaining data points.

In this paper, this algorithm is applied on three datasets namely WBC (9-D), Iris (4-D), and seeds (7-D) normalized in the interval of [0,1] from [2].

A. Class Separation for Iris (4-D) Dataset

The 4-D Iris data [2] consist of 150 cases with 3 classes (setosa, versicolor and virginica) with 50 cases each based on sepal and petal lengths and widths. Once the data are loaded to the interactive visualization software, all the four coordinate axes are checked for good vertical separation. This is obtained when the coordinate sequence is (SL, SW) and (PL, PW). For consistency, SL, SW, PL, PW coordinates are substituted with X_1, X_2, X_3 and X_4 for further analysis.

The set of *dominance areas* discovered by interactive and automatic means for the iris data classification are as follows:

$$R_1 = (x_4 < 0.21)$$

$$R_2 = R_{21} \,\&\, R_{22}$$

$$R_{21} = (x_3 < 0.71)$$

$$R_{22} = (x_4 < 0.7)$$

$$R_3 = R_{31} \text{ or } R_{32} \text{ or } R_{33}$$

$$R_3 = R_{31} \text{ or } R_{32} \text{ or } R_{33}$$

$$R_{31m} = [(0.7 < x_4 < 0.8) \,\&\, (0 \le x_1 \le 1)] \text{ (mix)}$$

$$R_{31p} = [(0 \le x_3 \le 1) \,\&\, (x_2 > 0.45)] \text{ (pure)}$$

$$R_{32} = R_{32m} \,\&\, R_{32p}$$

$$R_{32m} = [(0.55 < x_4 < 0.65 \,\&\, x_2 > 0.2) \,\&\, (0 \le x_1 \le 1)] \text{ (mix)}$$

$$R_{32p} = [(0 \le x_3 \le 1) \,\&\, (0.5 < x_3 < 0.67 \,\&\, x_1 > 0.2)] \text{ (pure)}$$

$$R_{33} = R_{33m} \,\&\, R_{33p}$$

$$R_{33m} = [(x_2 < 0.2 \,\&\, x_4 < 0.6) \,\&\, (0 \le x_1 \le 1)] \text{ (mix)}$$

$$R_{33p} = [(0 \le x_3 \le 1) \,\&\, (x_1 < 0.2 \,\&\, x_3 < 0.67)] \text{ (pure)}$$

(The concepts of mix and pure are defined below).

The classification rule in first iteration is:

$$\text{If } (x_4) \in R_1, \text{ then } \mathbf{x} \in \text{class 1}$$

The R_1 area used for class 1 separation is visualized in Fig. 2. After visualization of R_1, it was observed that a large part of the R_1 is empty. This would cause overgeneralization of the model [9]. To avoid this, the width of the rectangle is reduced and hence the modified dominance rectangle R'_1 is defined as:

$$R'_1 = (x_4 < 0.21 \,\&\, x_3 < 0.2)$$

The modified classification rule in the first iteration is:

$$\text{If } (x_4, x_3) \in R'_1, \text{ then } \mathbf{x} \in \text{class 1}$$

Visualization of R'_1 is displayed in Fig. 3. The cases that follow rule for R'_1 were classified into class 1. Remaining cases that do not follow R'_1 are separated using areas R_2 and R_3 in the second and third iteration, respectively. For the remaining data, the coordinates that has best vertical separation of the data are selected using observation. It is done using the SCPVis software with (X_1, X_3) and (X_2, X_4) pairs where X_3 and X_4 are vertical coordinates to get a better class separation. The rule for the classification in second iteration is:

$$\text{If } (x_1, x_2, x_3, x_4) \in R_2 \text{ then } \mathbf{x} \in \text{class 2, else } \mathbf{x} \in \text{class 3}$$

In the second iteration, we discover coarse areas $R_2 = R_{21} \& R_{22}$ which defines class 2, where R_{21} is a rectangle in (X_1, X_3) and R_{22} is a rectangle in (X_2, X_4).

In the third iteration, we refine R_2 and create a new area R_3 by discovering a "mix" (sub-rectangle) R_{31m} in rectangle R_{21} where the classes 2 and 3 overlap. Then we refine area R_2 by excluding R_{31m} from R_{21} and make a new rule as follows. We trace lines that go from the mix R_{31m} rectangle in search for a "pure" sub-rectangle R_{31p} of R_{22} where only lines from class 2 go or where class 2 dominates. This allows us to generate new area R_{31p} and a respective rule: if $\mathbf{x} \in R_{31m}$ & $\mathbf{x} \in R_{31p}$ then $\mathbf{x} \in$ class 2, where

$$R_{31m} = [(0.7 < x_4 < 0.8) \,\&\, (0 \le x_1 \le 1)] \text{ (mix)}$$

$$R_{31p} = [(0 \le x_3 \le 1) \,\&\, (x_2 > 0.45)] \text{ (pure)}$$

Similarly, we exclude pure sub-rectangle R_{31p} from R_{22} when R_2 is refined to R'_2

Similar pairs of rectangles (R_{32m}, R_{32p}), (R_{33m}, R_{33p}) and rules are generated for another refinement of R_2. The modified rule is recomputed as:

$$\text{If } (x_1, x_2, x_3, x_4) \in (R'_2 \text{ or } R_3), \text{ then } \mathbf{x} \in \text{class 2, else } \mathbf{x} \in \text{class 3}$$

The R'_2 and R_3 rules used for class 2 separation are visualized in Fig. 4. For easier understanding of the rules, the visualization in Fig. 4 shows cases that involve all rules. Also, the heights of the dominance rectangles R_{21} and R_{22} can be reduced, like in the step performed for R_1 to avoid overgeneralization as Fig. 3 shows.

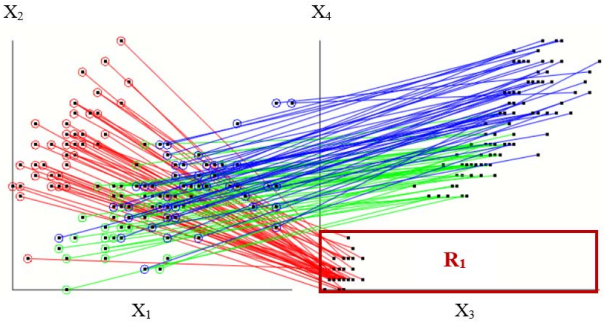

Fig. 2: Visualization of rule for R_1 on Iris dataset (4-D) for class 1 separation.

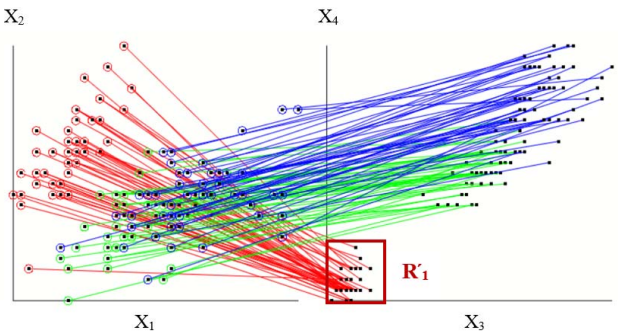

Fig. 3: Visualization of rule for R′1 on Iris dataset (4-D) for class 1 separation.

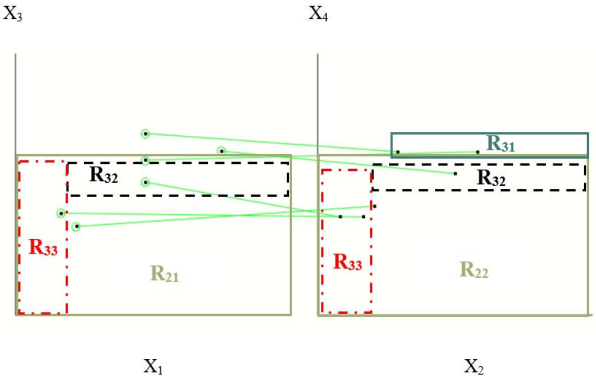

Fig. 4: Visualization of rules for R_2 and R_3 on Iris dataset (4-D) for class 2 separation with six instances from class 2.

Fig. 5 displays the rules with all instances for class 2 separation.

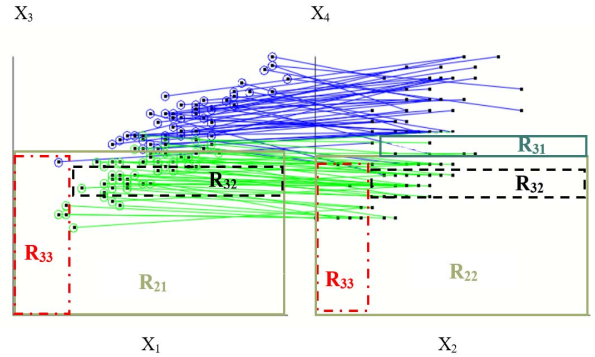

Fig. 5: Visualization of rules for R_2 and R_3 on Iris dataset (4-D) for class 2 separation with all instances from class 2 and class 3.

B. Class Separation for Wisconsin Breast Cancer(9-D) Dataset

Wisconsin Breast Cancer (WBC) dataset consists of 9 attributes and 699 instances where 16 instances were incomplete and hence were removed. Remaining 683 data points consists two classes of data: 444 benign cases and 239 malignant cases [2]. Upon loading to the SPCVis software, it is as displayed in Fig. 1. For this approach X_2 was duplicated as 10^{th} coordinate. The sequence of the coordinates is (X_9, X_1), (X_6, X_2), (X_4, X_3), (X_5, X_2) and (X_8, X_7). The accuracy obtained after 10-fold cross validation technique is **99.56%.** Below are the areas applied for classification of WBC dataset:

$$R_1 = (x_2 < 0.3 \ \& \ x_3 < 0.1)$$

$$R_2 = (x_2 > 0.4 \ \& \ x_7 > 0.4)$$

$$R_3 = (x_1<0.4)\&(x_4<0.4)\&(x_6<0.5)\&(x_9<0.3)\&(x_3<0.4)\&(x_8<0.3)$$

$$R_4 = (x_4 > 0.3 \ \& \ x_7 < 0.4 \ \& \ \& \ x_3 > 0.4 \ \& \ x_8 > 0.3)$$

$$R_5 = ((x_1 > 0 \ \& \ x_1 < 0.8) \ \& \ (x_5 < 0.5 \text{ or } (x_5 > 0.6 \ \& \ x_5 < 1)) \ \& \ (x_2 < 0.4 \text{ or } x_2 > 0.8) \ \& (x_6 < 0.3 \text{ or } (x_6 > 0.4 \ \& \ x_6 < 0.8)))$$

$$R_6 = R_{61} \text{ or } R_{62}$$

$$R_{61} = [x_7 < 0.6 \ \& \ (x_4 < 0.1 \text{ or } (x_4 > 0.2 \ \& \ x_4 < 0.3))] \text{ or}$$

$$R_{62} = [(x_4 > 0.4 \ \& \ x_4 < 0.6)) \ \& \ (x_8 <0.3 \text{ or } (x_8 > 0.4 \ \& \ x_8 < 0.7))]$$

$$R_7 = (0.3< x_1< 0.7)\&[((x_2<0.9)) \ \&(x_6<0.1 \text{ or } (x_6>0.2\&x_6< 0.6)]$$

$$R_8 = (x_3 < 0.2 \text{ or } (0.3 < x_3 < 0.5) \text{ or } x_3 > 0.6) \ \& \ (x_4 < 0.5 \text{ or } x_4 > 0.7) \ \& \ (x_8 < 0.5 \text{ or } x_8 > 0.7)$$

$$R_9 = [(x_6 > 0.6 \ \& \ ((x_2 >0.2 \ \& \ x_2 < 0.6) \text{ or } (x_2 > 0.6 \ \& \ x_2 < 0.8)) \ \& \ (x_3 < 0.5 \ \& \ (0.6 < x_9 < 0.8))]$$

$$R_{10} = [((x_5 < 0.5 \text{ or } (0.6 < x_5 < 0.7)) \ \& \ ((x_4< 0.2 \text{ or } (0.4 < x_4 < 0.5) \text{ or } x_4 > 0.9) \ \& \ (x_7 < 0.3 \text{ or } x_7 > 0.6)) \ \& \ x_8 < 0.8)]$$

The rules for the classification in first iteration are:

$$\text{If } (x_2, x_3) \in R_1, \text{ then x} \in \text{class 1}$$

$$\text{If } (x_2, x_7) \in R_2, \text{ then x} \in \text{class 2}$$

The cases that followed rule for R_1 were classified as class 1(malignant) and the cases that followed rule for R_2 were classified to class 2. The cases that followed neither of the rules were separated in the second iteration. The best fit sequence for the second iteration was (X_1, X_4), (X_6, X_9), (X_7, X_3), (X_5, X_8) and (X_2, X_9). In the second iteration, X_9 was duplicated as 10^{th} coordinate. The criteria for the classification in the second iteration are:

$$\text{If } (x_1, x_4, x_6, x_9, x_3, x_8) \in R_3, \text{ then x} \in \text{class 1}$$

$$\text{If } (x_3, x_4, x_8) \in R_4, \text{ then x} \in \text{class 2}$$

The cases that followed rule for R_3 were classified as class 1(malignant) and the cases that followed rule for R_4 were classified as class 2. Rest of the cases that followed neither of the rules were separated in the third iteration. The best fit sequence for the third iteration is (X_1, X_5), (X_2, X_6), (X_3, X_9), (X_4, X_7) and (X_8, X_9). In the third iteration, X_9 was duplicated as 10^{th} coordinate. The rules for the classification in the third iteration are:

$$\text{If } (x_1, x_5, x_2, x_6, x_7, x_4, x_8) \in (R_5 \ \& \ R_6), \text{ then x} \in \text{class 1}$$

$$\text{If } (x_1, x_2, x_6, x_3, x_4, x_8) \in (R_7 \ \& \ R_8), \text{ then x} \in \text{class 2}$$

The cases that followed rules for R_5 and R_6 on (x_1, x_5, x_2, x_6, x_7, x_4, x_8) were classified as class 1(benign) and that followed rules for R_7 and R_8 on (x_1, x_2, x_6, x_3, x_4, x_8) were classified as class 2 (malignant). Rest of the cases that followed neither of the rules were separated in the fourth iteration. The best fit sequence for the third iteration is (X_2, X_1), (X_3, X_5), (X_4, X_8), (X_7, X_6) and (X_9, X_8). In the fourth iteration, X_8 is duplicated as 10^{th} coordinate.

The rules for the classification in first iteration are:

$$\text{If } (x_2, x_1, x_5, x_4, x_8, x_7) \in (R_9 \ \& \ R_{10}), \text{ then x} \in \text{class 1,}$$
$$\text{else x} \in \text{class 2}$$

The cases that followed rules for R_9 and R_{10} on (x_1, x_5, x_2, x_6, x_7, x_4, x_8) belonged to class 1 and the rest belonged to class 2. Fig. 6 visualizes areas R_5 and R_6 with three instances from class 1.

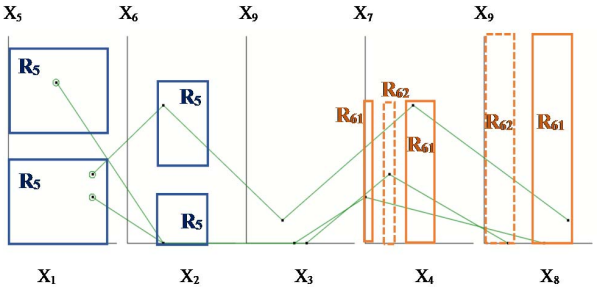

Fig. 6: Visualization of rules for R_5 and R_6 on Breast Cancer dataset (9-D) for class 1 separation with 3 instances from class. 1

Fig. 7 visualizes areas R_5 and R_6 on WBC dataset including all the instances from class 1.

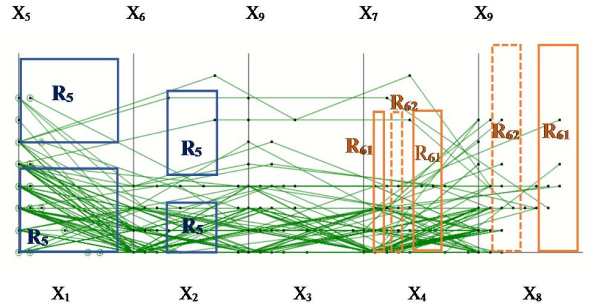

Fig. 7: Visualization of rules for R_5 and R_6 on WBC dataset (9-D) for class 1 separation with all the instances from class 1.

Fig. 8 visualizes rules for R_1 and R_{10}, Fig. 9 visualizes rules for R_3 and R_9 and Fig. 10 visualizes rules for R_7 and R_8 on WBC dataset including 5 instances from class 2.

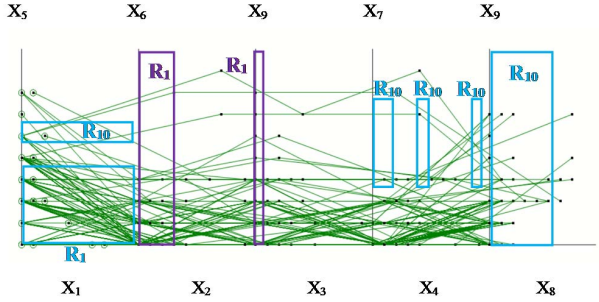

Fig. 8: Visualization of rules for R_1 and R_{10} on WBC (9-D) for class 1 separation with all the instances from class 1.

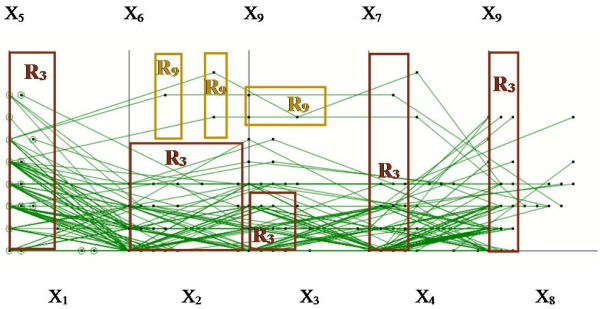

Fig. 9: Visualization of rules for R_3 and R_9 on WBC dataset (9-D) for class 1 separation with all the instances from class 1.

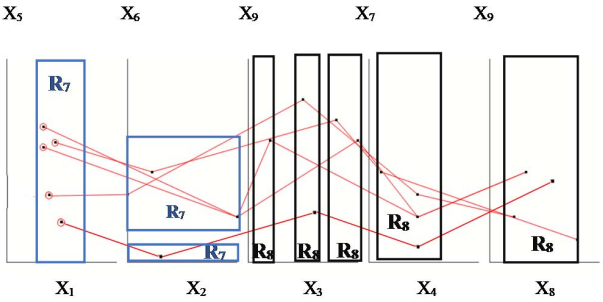

Fig. 10: Visualization of rules for R_7 and R_8 on WBC dataset (9-D) for class 2 separation with 5 instances from class 2.

Figs. 10 and 11 visualize rules for R_7 and R_8 on WBC dataset from class 2 for selected and all cases, respectively.

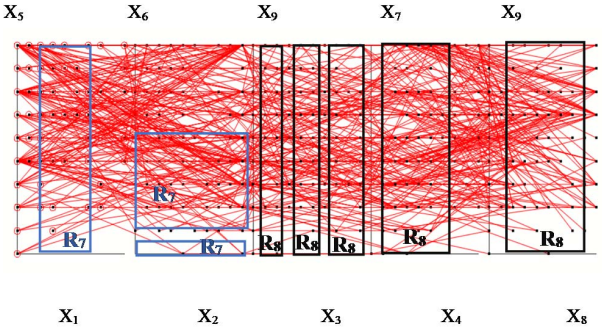

Fig. 11: Visualization of Rules 7 and 8 on WBC dataset (9-D) for class 2 separation with all the instances from class 2.

Fig. 12 visualizes rules for R_2 and R_4 on WBC dataset including all the instances from class 2.

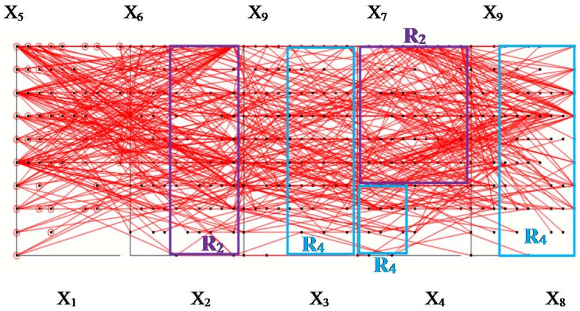

Fig. 12: Visualization of rules for R_2 and R_4 on Breast Cancer dataset (9-D) for class 2 separation with all the instances from class 2.

While it appears from Figs. 8, 9 and 10 that the dominance rectangles for R_{10}, R_3 and R_7 overlap with one another in (X_1, X_5), the start and end points of the data used in these dominance rectangles are completely different. For instance, the dominance rectangles for areas R_{10} in Fig. 8 start in (X_1 , X_5), pass through (X_4 , X_7) and end in (X_8 , X_9). The dominance rectangles for areas R_3 in Fig. 9 start in (X_1 , X_5), pass through (X_2 , X_6), (X_3 , X_9), (X_4 , X_7) and end in (X_8 , X_9). The dominance rectangles for R_7 in Fig. 10 start in (X_1 , X_5) and end in (X_2 , X_6).

C. Class Separation for Seeds (7-D) Dataset

The 7-D seeds data from UCI Machine Learning Repository [2] consist of 210 cases and 3 classes (Kama, Rosa and Canadian) with 70 cases each based on geometric perimeter of

the wheat kernels. Once the data are loaded to the interactive visualization software, all the seven coordinate axes are checked for good vertical separation. X_2 coordinate is duplicated as the 8th coordinate. The coordinate sequence obtained is (X_3, X_1), (X_4, X_2), (X_6, X_5) and (X_7, X_2).

The dominance areas for seeds classification are as follows:

$R_1 = [((0.451 < x_1 < 0.463)$ or $(x_1 > 0.469)) \,\&\, ((0.306 < x_7 < 0.453)$ or $x_7 > 0.840)]$

$R_2 = R_{21}$ or R_{22}

$R_{21} = [(x_4 < 0.428$ or $x_4 > 0.457) \,\&\, x_2 < 0.522]$

$R_{22} = [((0.557 < x_2 < 0.614)$ or $x_2 > 0.639) \,\&\, x_5 > 0.4]$

$R_3 = [((0.139 < x_1 < 0.169)$ or $(x_1 > 0.2)) \,\&\, (x_6 < 0.282)]$

$R_4 = [(0.16 < x_2 < 0.178)$ or $(0.21 < x_2 < 0.279)$ or $x_2 > 0.491)$ $\&\, (x_4 < 0.270$ or $(x_2 > 0.491 \,\&\, x_4 > 0.672)]$

$R_5 = [(x_5 > 0.455 \,\&\, x_6 > 0.349) \,\&\, (x_3 < 0.475$ or $x_3 > 0.507)]$

The rule for classification in the first iteration is:

If $(x_1, x_7, x_4, x_2, x_6) \in (R_1$ or $R_2)$, then **x** ∈ class 2

The instances that follow rules for R_1 and R_2 on (x_1, x_7, x_4, x_2, x_6) were classified into class 2. The rest of the instances that did not follow the rules were separated in iteration 2. Fig. 13 visualizes rules for R_1 and R_2 used for class 2 separation with eight sample instances from class 2.

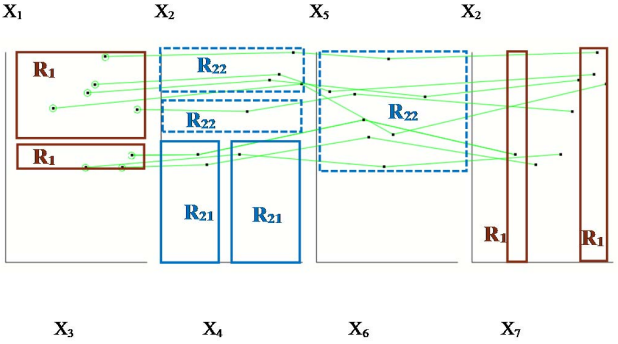

Fig. 13: Visualization of rules for R_1 and R_2 on Seeds dataset (7-D) or class 2 separation with eight instances from class 2.

Fig. 14 visualizes rules for R_1 and R_2 used for class 2 separation with all instances from class 2.

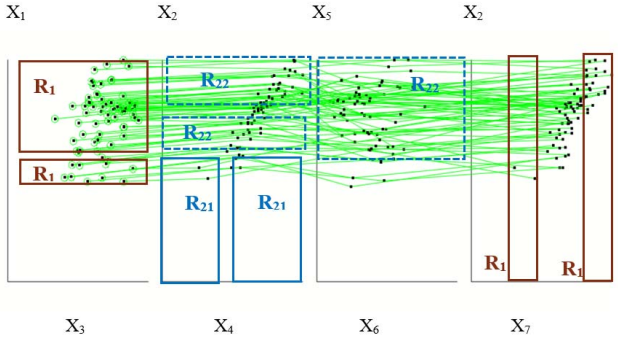

Fig. 14: Visualization of rules for R_1 and R_2 on Seeds dataset (7-D) for class 2 separation with all instances from class 2.

For the second iteration, X_1 coordinate is duplicated as the 8th coordinate. The coordinate sequence obtained is (X_3, X_1), (X_4, X_2), (X_6, X_5) and (X_7, X_1).

If $(x_1, x_6, x_2, x_4, x_5, x_3) \in F(R_3, R_4, R_5)$, then x ∈ class 1, else x ∈ class 3,

where F represent combinations of R_3, R_4, R_5 captures in Fig. 15. Instances that followed rules for R_3, R_4 and R_5 are classified into class 2 and the rest of the instances that failed to follow rules for R_3, R_4 and R_5 are classified to class 3. Fig. 15 visualizes rules for R_3, R_4 and R_5 used for class 1 separation with eight instances from class 1.

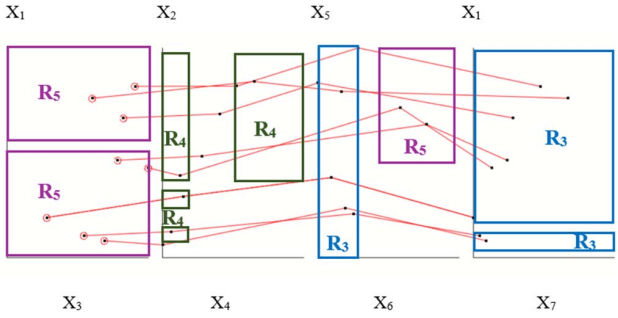

Fig. 15: Visualization of rules for R_3, R_4 and R_5 on Seeds dataset (7-D) for class 1 separation with eight instances from class 1.

Fig. 16 visualizes rules for R_3, R_4 and R_5 used for class 1 separation with all instances from class 1.

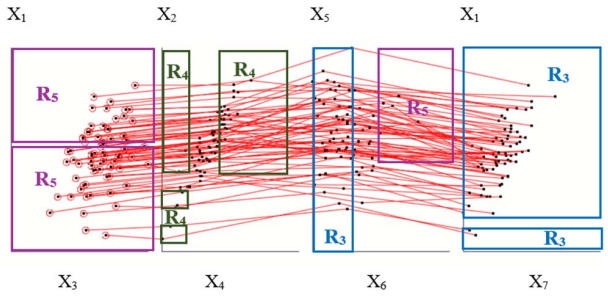

Fig. 16: Visualization of rules for R_3, R_4 and R_5 on Seeds dataset (7-D) for class 1 separation with all instances from class 1.

From Figs. 14 and 16, the dominance rectangles appear to overlap with one another. As discussed earlier for breast cancer dataset, similar condition applies to this dataset as well, indicating that the start and end points of the data defined in these dominance rectangles are completely different. Also, the heights of rectangles defined by R_{21} in (X_4, X_2) and R_1 in (X_7, X_2) can be reduced to avoid overgeneralization [9].

IV. Sumamry of Experimental Results and Comparison With Published Results

The accuracy of the data separation is computed using 10-fold cross validation. Instead of using random splits, the worst-case heuristics [8] are used in the validation split. The worst split data can be easily obtained with the help of the Interactive SPCVis.

The first validation split contains the worst split data followed by the rest validation splits. The accuracy for data separation for all the three dataset is tabulated in Table 1.

Table 1: 10-fold cross validation (with worst case heuristics).

	Iris Data (4-D)	Breast Cancer Data(9-D)	Seeds Data(7-D)
1	100	95.65	100
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100
7	100	100	100
8	100	100	100
9	100	100	100
10	100	100	100
Average	**100**	**99.56**	**100**

So far, the best results obtained for WBC data is by DCP/RPPR which combines two interpretable algorithms is 99.3 % [11]. Other accuracies include 96.995 % [1] and 97.28% [13] using non interpretable ML algorithms like SVM [1] and a combination of SVM, C4.5 and kNN and Bayesian algorithms [13]. With the combination of interactive data visualization and Iterative Logical Classifier Algorithm, the accuracy obtained was 99.56% that outperformed the results in [1, 11, 13]. The accuracy of 4-D iris data classification as seen in [6] is 100%. The technique used is multilayer visual knowledge discovery. Also, with black-box models, the accuracies obtained are 98.67% using simple k-Means and J48 classifier [10], 96.66% using neural networks [14]. Using the technique proposed in this paper, the accuracy obtained was 100% that outperformed all the results using black box classification models in [10, 14].

Table 2: Comparison of Different Classification Models.

Classification Algorithms	Accuracy %
Breast Cancer data (9-D)	
Iterative Logical Classifier	**99.56**
SVM [1]	96.995
DCP/RPPR [11]	99.3
SVM/C4.5/kNN/Bayesian [13]	97.28
Iris Data(4-D)	
Iterative Logical Classifier	**100**
Multilayer Visual Knowledge discovery [6]	100
k-Means +J48 classifier [10]	98.67
Neural Network [14]	96.66
Seeds Data(7-D)	
Iterative Logical Classifier	**100**
Deep Neural Network [3]	100
K- nearest neighbor [12]	95.71

The seeds dataset accuracy using K-nearest neighbor is 95.7143% [12] and 100% using Deep Neural Networks [3]. With the combination of interactive data visualization and Iterative Logical Classifier Algorithm, the accuracy obtained was 100% which is better than the traditional ML models in [3, 12]. The accuracies are summarized in table 2.

V. CONCLUSION

This paper demonstrates that data visualization combined with self-service or democratized machine learning implemented in the form of the Iterative Logical Classifier algorithm can compete and outperform the traditional black-box machine learning classifier models. This algorithm and SPCVis generate a set of rules that classify the data. If some of the data points fail to follow the rules, they are processed at the subsequent iterations where rules are refined and tested on the dataset. This process is continued until all the data points are classified, thereby resulting in 100% coverage of the data. However, SPC visualization captures only specific types of patterns in the data and the current SPCVis implementation is mainly focused on the interactive rule discovery with limited automation. The future work is to use other General Line Coordinates visualizations [6,7] for classification with more options for interactive controls and automation with minimum overgeneralization.

REFERENCES

[1] Christobel, A. and Y. Sivaprakasam. An empirical comparison of data mining classification methods. International Journal of Computer Information Systems 3.2 (2011): 24-28.

[2] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[3] Eldem A. An Application of Deep Neural Network for Classification of Wheat Seeds. Avrupa Bilim ve Teknoloji Dergisi.(19):213-20.

[4] Feurer M, Klein A, Eggensperger K, Springenberg JT, Blum M, Hutter F. Auto-sklearn: efficient and robust automated machine learning. In Automated Machine Learning 2019 (pp. 113-134). Springer, Cham.

[5] Hutter F, Kotthoff L, Vanschoren J. Automated machine learning: methods, systems, challenges. Springer Nature; 2019.

[6] Kovalerchuk B. Visual Knowledge Discovery and Machine Learning, Springer, 2018.

[7] Kovalerchuk, B. and A. Gharawi. "Decreasing Occlusion and Increasing Explanation in Interactive Visual Knowledge Discovery." International Conference on Human Interface and the Management of Information. Springer, Cham, 2018.

[8] Kovalerchuk, B. Enhancement of Cross Validation Using Hybrid Visual and Analytical Means with Shannon Function. Beyond Traditional Probabilistic Data Processing Techniques: Interval, Fuzzy etc. Methods and Their Applications. Springer, Cham, 2020. 517-543.

[9] Kovalerchuk B, Grishin V. Reversible Data Visualization to Support Machine Learning. In International Conference on Human Interface and the Management of Information 2018 Jul 15 (pp. 45-59). Springer, Cham.

[10] Kumar, V. and N. Rathee. Knowledge discovery from database Using an integration of clustering and classification. International Journal of Advanced Computer Science and Applications 2.3 (2011): 29-33.

[11] Neuhaus N, Kovalerchuk B. Interpretable Machine Learning with Boosting by Boolean Algorithm. In2019 Joint 8th ICIEV and 2019 3rd icIVPR, 2019 May 30 (pp. 307-311). IEEE.

[12] Sabancı K, Akkaya M. Classification of Different Wheat Varieties by Using Data Mining Algorithms. International Journal of Intelligent Systems and Applications in Engineering. 2016 May 27;4(2):40-4.

[13] Salama GI, Abdelhalim M, Zeid MA. Breast cancer diagnosis on three different datasets using multi-classifiers. Breast Cancer (WDBC). 2012 Jan;32(569):2.

[14] Swain M, Dash SK, Dash S, Mohapatra A. An approach for iris plant classification using neural network. International Journal on Soft Computing. 2012 Feb 1;3(1):79.