Spring 2021

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

Sridevi Narayana Wagle
Central Washington University, sridevi.wagle@cwu.edu

INTERACTIVE VISUAL SELF-SERVICE DATA CLASSIFICATION APPROACH TO DEMOCRATIZE MACHINE LEARNING

A Thesis

Presented to

The Graduate Faculty

Central Washington University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

Computational Science

---

by

Sridevi Narayana Wagle

June 2021

CENTRAL WASHINGTON UNIVERSITY

Graduate Studies

We hereby approve the thesis of

Sridevi Narayana Wagle

Candidate for the degree of Master of Science

APPROVED FOR THE GRADUATE FACULTY

_____ _____
Dr. Boris Kovalerchuk

_____ _____
Dr. Razvan Andonie

_____ _____
Dr. Szilard Vajda

_____ _____
Dean of Graduate Studies

ABSTRACT

INTERACTIVE VISUAL SELF-SERVICE DATA CLASSIFICATION

APPROACH TO DEMOCRATIZE MACHINE LEARNING

by

Sridevi Narayana Wagle

June 2021

Machine learning algorithms often produce models considered as complex black-box models by both end users and developers. Such algorithms fail to explain the model in terms of the domain they are designed for. The proposed *Iterative Visual Logical Classifier* (IVLC) is an interpretable machine learning algorithm that allows end users to design a model and classify data with more confidence and without having to compromise on the accuracy. Such technique is especially helpful when dealing with sensitive and crucial data like cancer data in the medical domain with high cost of errors.

With the help of the proposed interactive and lossless multidimensional visualization, end users can identify the pattern in the data based on which they can make explainable decisions. Such options would not be possible in black box machine learning methodologies. The interpretable IVLC algorithm is supported by the *Interactive Shifted Paired Coordinates Software System* (SPCVis). It is a lossless multidimensional data visualization system with interactive features. The interactive approach provides flexibility to the end user to perform data classification as self-service without having to rely on a machine learning expert.

Interactive pattern discovery becomes challenging while dealing with large datasets with hundreds of dimensions/features. To overcome this problem, an automated classification approach combined with new *Coordinate Order Optimizer* (COO) algorithm and a *Genetic algorithm* (GA) is proposed. The COO algorithm automatically generates the coordinate pair sequences that best represent the data separation and GA helps optimizing the proposed IVLC algorithm by automatically generating the areas for data classification. The feasibility of the approach is shown by experiments on benchmark datasets covering both interactive and automated processes used for data classification.

ACKNOWLEDGMENTS

With deep gratitude and sincerity, I would like to thank my advisor Dr. Boris Kovalerchuk for his continuous support and guidance throughout this thesis work. I would also like to thank Drs. Razvan Andonie and Szilard Vajda for their valuable feedback, as well as my colleagues for their help and assistance during the graduate course. Finally, I would like to thank my family and friends for their unconditional support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (CONTINUED)

LIST OF FIGURES (CONTINUED)

CHAPTER I

INTRODUCTION

The exponential growth in machine learning has resulted in smart applications making critical decisions without human intervention. However, people with no technical background find it often difficult to rely on machine learning models, since many of them are black-boxed [8].

Interpretability plays a crucial role in deciphering behind-the-scenes actions in a machine learning algorithm. These interpretable machine learning techniques when combined with data visualization and analytics unfold numerous ways to discover deep hidden patterns in the data [11]. Interpretable machine learning models are transparent clearly explaining the technique behind the model prediction. This gives more clarity to the end user who can rely on the model with more confidence compared to black-box machine learning models [9].

When it comes to data visualization, most of the visualization techniques used to display multidimensional data like Principal Component Analysis (PCA), Heat maps, Self-organizing maps etc. are lossy and irreversible [10]. With the help of new lossless data visualization, it is now possible to maintain structural integrity of the data. Moreover, these lossless techniques are completely reversible [10, 11].

In our proposed approach, we focus on interpretable data classification techniques using a combination of interactive data visualization and analytical rules. The Shifted

Paired Coordinates (SPC) system [11] is a lossless and a more compact way to visualize multidimensional data when compared to other lossless data visualization like Parallel Coordinates [10]. SPC allows discovering patterns more efficiently since the number of lines required to display the data is reduced by half [11].

However, as the size of the data increases, pattern discovery becomes challenging due to occlusion. To reduce occlusion and expose hidden patterns, data representation needs to be reorganized. This can be achieved by applying interactive techniques such as changing the order of coordinates, swapping within the coordinate pair etc. These interactive capabilities allow the end-users to intervene and optimize the classification model generated by the machine thereby improving the overall model performance. To further leverage the classification technique, areas within the coordinate pairs are discovered either interactively or automatically, unravelling the deep hidden patterns in the data. Analytical rules are then built on the areas discovered to classify the data.

The data classification approach using analytical rules is implemented using IVLC algorithm [23] wherein the areas and analytical rules are generated in iterations until all the data are covered. This implementation works for smaller datasets, the areas and analytical rules can be generated interactively by the end users. We expand this work on large datasets by adding automation and more visual operations.

For the classification of larger data with hundreds of dimensions, interactive methods alone do not suffice. Due to high occlusion for such datasets, discovery of patterns purely with interactive methods becomes challenging where deep hidden patterns

cannot be detected by humans, thereby leading to the necessity of automation. In our work, automation for the classification approach is implemented in two stages:

First Stage: Order of coordinates are optimized using COO algorithm. It is used to find the best coordinate orders for the Shifted Paired Coordinates System where the data separation can be visually identified along the vertical axis of each coordinate pair.

Second Stage: This stage involves generating the areas within the coordinate pairs with close *Proximity* and high *Purity* or fitness i.e., areas that are closer to each other with high density of data belonging to same class are generated. GA is used for the automatic generation of areas with high purity. It involves generation of random areas and that are mutated and altered over generations to obtain the areas with high *Purity* or fitness [1].

Several experiments are conducted on benchmark datasets like Wisconsin Brest Cancer (WBC), Iris, Seeds and Air Pressure System (APS) in Scania Trucks. The experiments are performed using both interactive and automated approach using 10-fold cross validation with worst case heuristics [13] where the initial validation set contains the data from one class overlapping with another class and vice versa. The results obtained with both interactive and automated techniques were on par with the published results in other studies.

Related Work

The Shifted Paired Coordinates (SPC) [11] visualize data losslessly using coordinate pairs where each pair of data is represented on a two dimensional plane. Figure 1 represents an 8D data (8, 1, 3, 9, 8, 3, 2, 5) using SPC, where pair (8,1) is visualized in (X_1, X_2), the next pair (3, 9) is visualized in (X_2, X_4) and so on. Then these points are connected to form a graph.

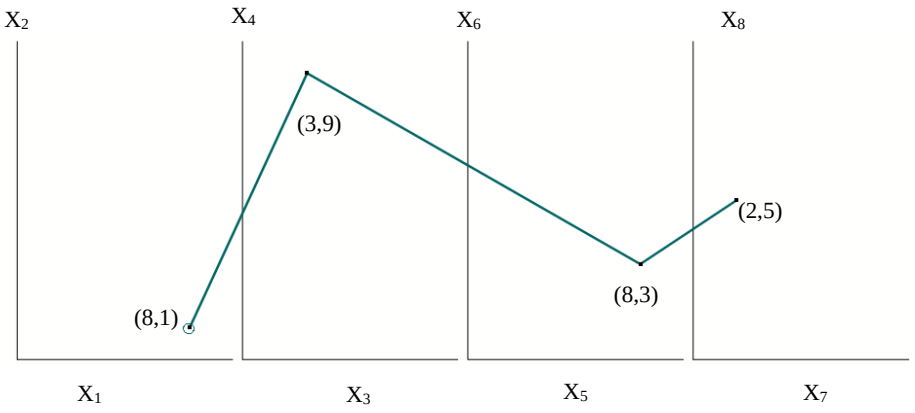
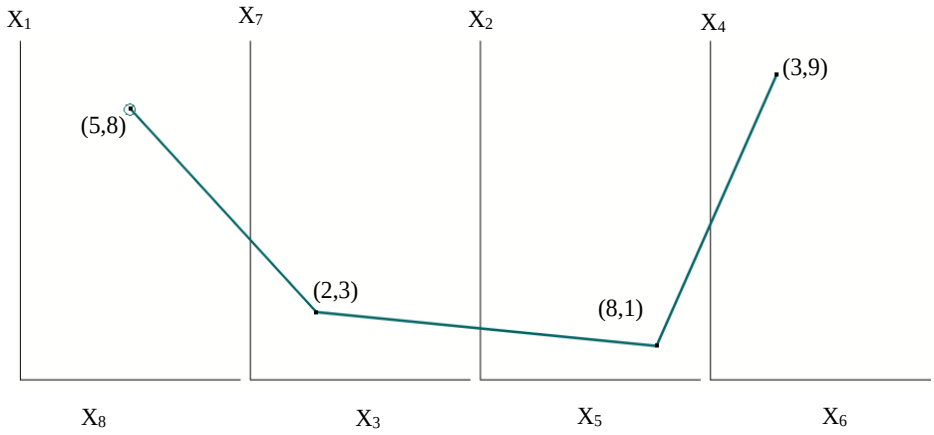


FIGURE 1: Representation of 8D data (8, 1, 3, 9, 8, 3, 2, 5) in SPC.

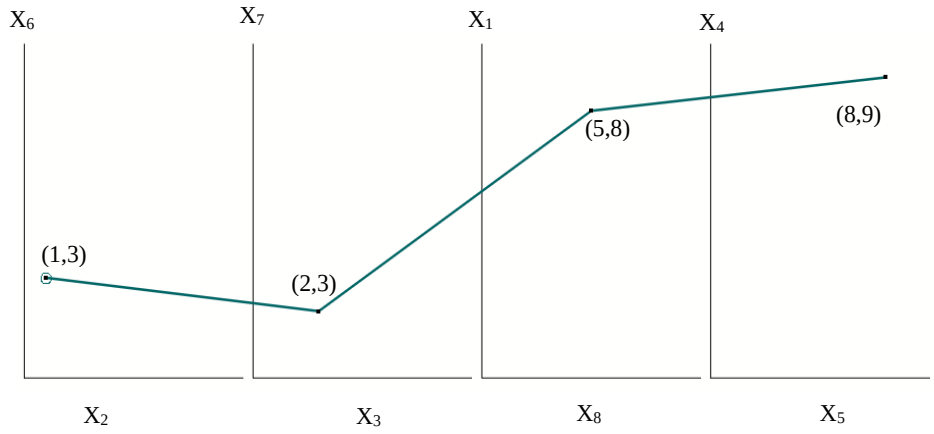
Same data can be displayed in multiple ways using different combinations of coordinate pairs. Figures 2a and 2b represent the same data with (X_8, X_1), (X_3, X_7), (X_5, X_2) and (X_6, X_4) sequence of coordinates and (X_2, X_6), (X_3, X_7), (X_8, X_1) and (X_5, X_4) sequence of coordinates, respectively.

Although SPC provides lossless data visualization, discovering patterns in the data becomes challenging due to occlusion. To overcome this challenge, IVLC algorithm

is used comprising of interactive controls to reorient the data by interactively changing the coordinate order to make the pattern discovery easier along with analytical rules to classify the data.



(a) 8D data with (X_8, X_1), (X_3, X_7), (X_5, X_2) and (X_6, X_4) coordinate pair sequence.



(b) 8D data with (X_2, X_6), (X_3, X_7), (X_8, X_1) and (X_5, X_4) coordinate pair sequence.

FIGURE 2: Representation of 8D data (8, 1, 3, 9, 8, 3, 2, 5) in SPC with different coordinate pair sequences.

Data classification using a combination of Shifted Paired Coordinates and analytical rules has already been implemented in [12]. The pattern discovery is simplified using FSP (Filter Search Present) algorithm wherein the algorithm filters out less efficient rules, searches for sequence of pairs of coordinates and presents the SPC visualizations. The FSP algorithm generates random sequences of pairs of coordinates to represent n-D data in SPC. Classification rules are discovered using high precision, recall and accuracy and allowing the end user to manually control the output rectangles produced by the algorithm.

In paper [12], the experiment was conducted on three datasets from UCI Machine learning repository namely, 9D Wisconsin Breast Cancer (WBC) data, 34D Ionosphere dataset and 8D Abalone dataset. The accuracies obtained were 93.60%, 98.78% and 98.60% respectively with 70%:30% of the data into training and test split. Although, the techniques used in [12] are visual, interpretable, and explainable to the domain expert, it lacks interactive controls like moving the graph, data reversing, non-linear scaling, non-orthogonal coordinates etc. Also, the coverage of the data is not 100%. All these shortcomings are addressed in our proposed data classification techniques.

In paper [2], GA is implemented in a radial visualization to optimize the visualization quality. Here, the algorithm is used as a parallelized random search procedure to generate set of POIs (Points of Interests). The overall algorithm is defined by generating an initial population where each individual is evaluated by the cost function based on Kruskal's stress. The next step is random parent selection where the best individual is selected out of the random selected individuals with advantage given to

the individuals with less POIs. Next step is applying a uniform crossover operator in such a way that the offspring has equal probability of selecting a gene from either of its parents. After crossover mutation operator is applied and cost evaluation is performed.

In our proposed approach, we use interactive and interpretable data classification technique to overcome the shortcomings in [12]. Our proposed IVLC algorithm discovers the classification rules with 100 % data coverage with interactive features. The proposed COO algorithm generates the optimized order of coordinate pairs based on statistical parameters rather than generating them randomly, as implemented in [12]. We use GA in a similar manner as in [2] to optimize the area generation in automatic data classification approach where we generate Areas of Interests (AOIs) rather than POIs. The best individual is selected from the population based on high *Purity* and close *Proximity*. We further build analytical rules on these selected AOIs for classifying the data.

This thesis offers a detailed explanation of the following:

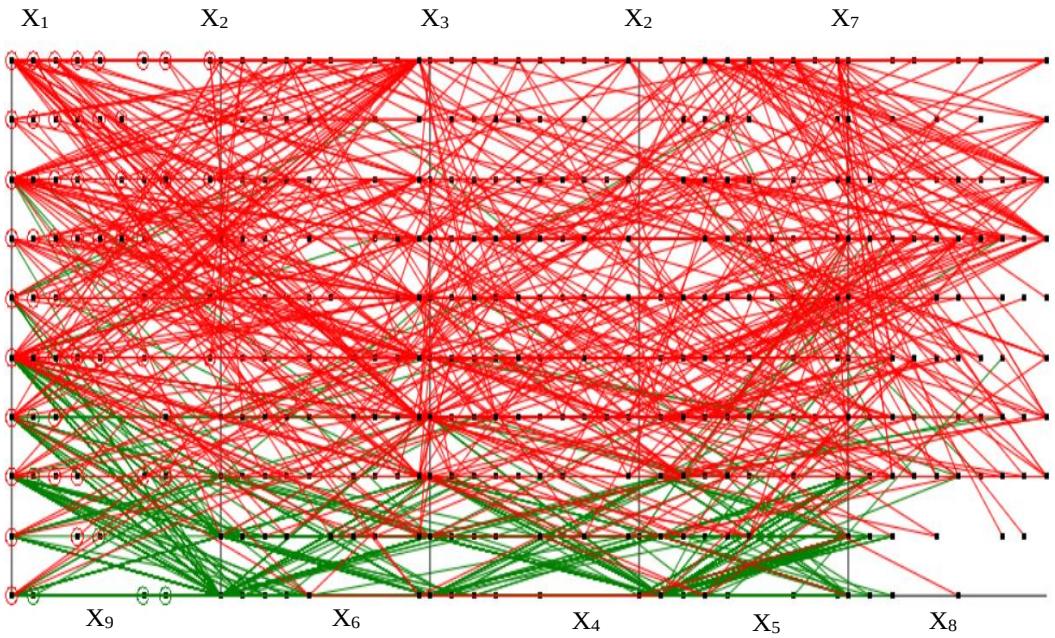
- Interactive Shifted Paired Coordinate Visualization System (SPCVis) [23]
- Different interactive techniques used in SPCVis
- Iterative Visual Logical Classifier (IVLC) algorithm
- Automation of the classification technique using:
    - Coordinate Order Optimizer (COO)
    - Genetic Algorithm (GA)
- Experiments with benchmark datasets
- Comparing the results with published results
- Conclusions

CHAPTER II

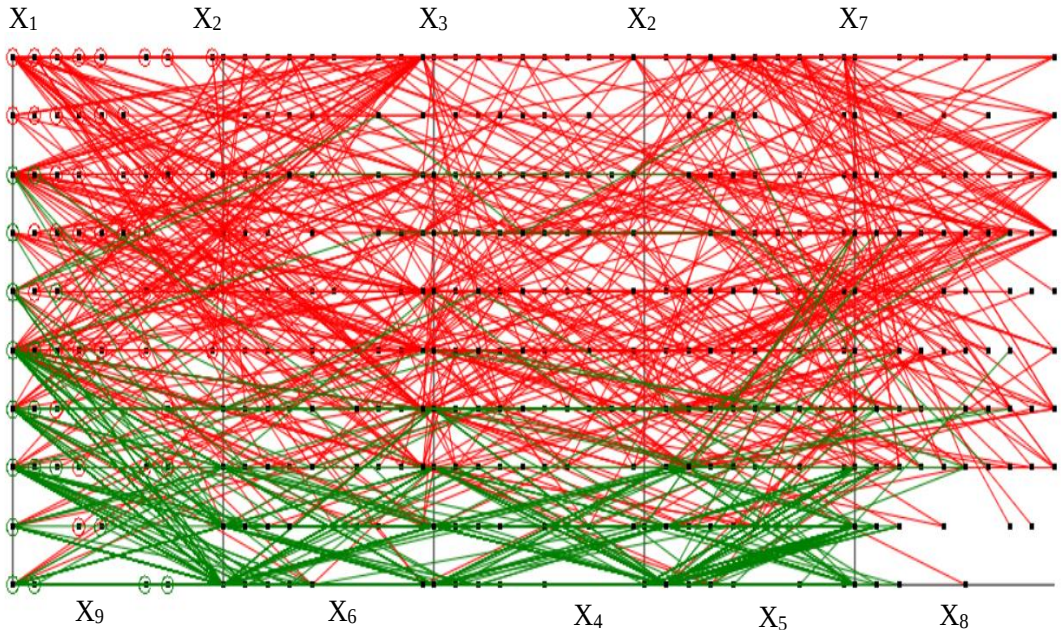
INTERACTIVE SHIFTED PAIRED COORDINATE SYSTEM

The lossless n-D visualization is achieved by representing data in *Interactive Shifted Paired Coordinate System* (SPCVis). Reordering the coordinates is one of the interactive features provided by the SPCVis software system. Using this feature, the coordinates are reordered in such a way that the class separation is prominent along the vertical coordinates. The discovering of coordinates to get good separation of classes is performed interactively by the user. The data used for SPCVis are normalized to [0, 1].

There are several interactive features provided to the end user like reversing data, non-linear scaling etc. For instance, if $\mathbf{x}$ is an n-D point where $\mathbf{x} = (x_1, x_2, x_3, \ldots x_n)$, the reverse of x_1 would display the data as $(1 - x_1, x_2, x_3, \ldots x_n)$. Also, SPCVis software system provides the user ability to click and drag the whole (X_i, X_j) plot to a desired location until occlusion is reduced. Another interactive control allows a user to display the user selected class on top of another class. Figure 3a displays WBC breast cancer data with red class on top and Figure 3b with green class on top. This helps the user to observe the pattern of individual classes more clearly.



(a) WBC data with red class on top.



(b) WBC data with green class on top.

FIGURE 3: WBC data (9D) visualized in SPCVis.

Non-linear scaling is an interactive feature provided by the SPCVis software where only a part of the user selected coordinates is scaled differently. The generalized formula for an n-D point x_j is given in Equation 2.1, where k is a constant and $0 < k < 1$, r is the resolution of the data i.e., the shortest distance between the data points. The value of k is set by the user. Figure 4 displays the WBC data after applying the non-linear scaling at $r = 0.1$, $k = 0.6$ on X_1 and $k = 0.3$ on X_2, X_5 and X_7 coordinates.

$$x_j' = \begin{cases} x_j\,, & \text{if } x_j < k \\ x_j + r \times graphWidth, & \text{if } k \leq x_j < 1 \end{cases} \tag{2.1}$$

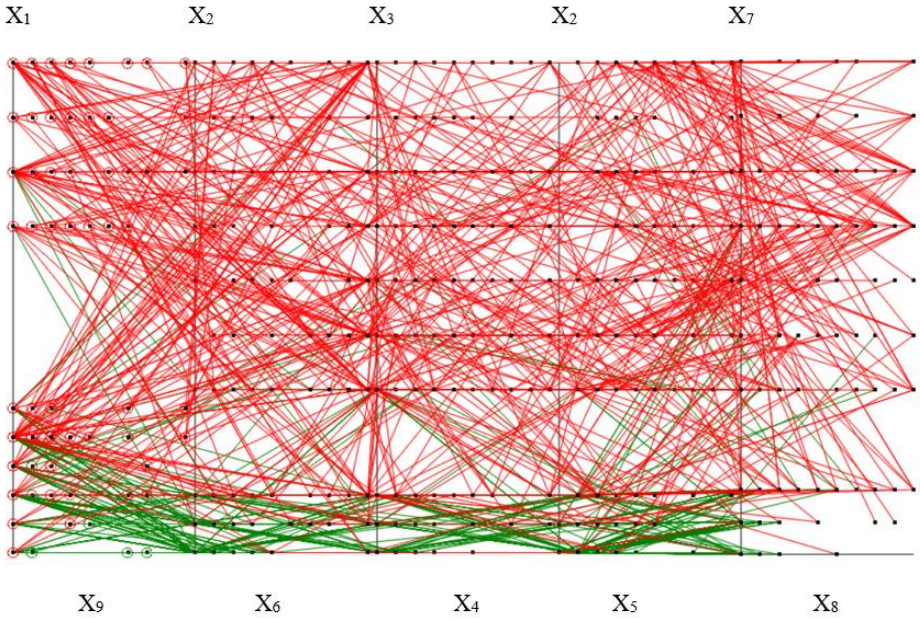


FIGURE 4: WBC data after non-linear scaling on all the vertical coordinates.

Another interactive feature provided by SPCVis software is Non-orthogonal coordinate system that has a coordinate inclined at an angle other than 90° with respect to the other coordinate. Figure 5 displays a simple 2D graph with Y coordinate inclined at

an angle of 30° with respect to its previous Y coordinate. Figure 6 displays Non-orthogonal coordinate representation with horizontal coordinates X_6 and X_5 inclined at -30°.

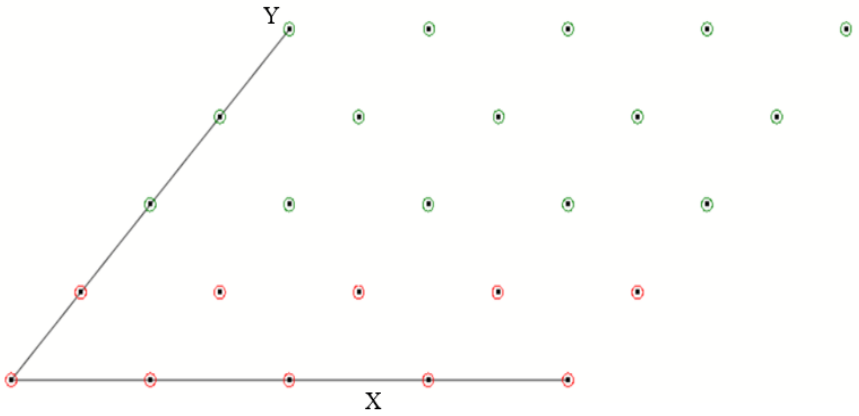


FIGURE 5: Non-orthogonal display of 2D data (Y=30°).

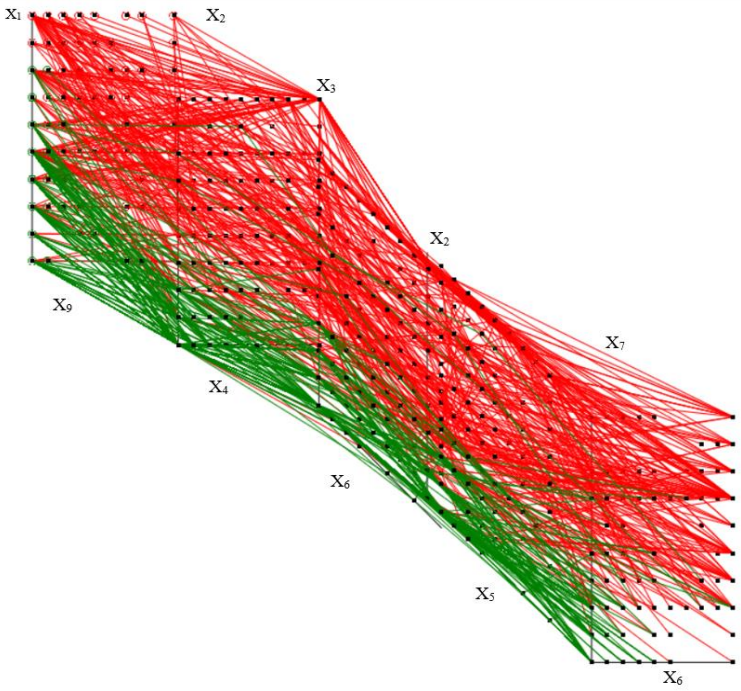


FIGURE 6: Non-orthogonal display of WBC data (X_6 and X_5 inclined at -30°).

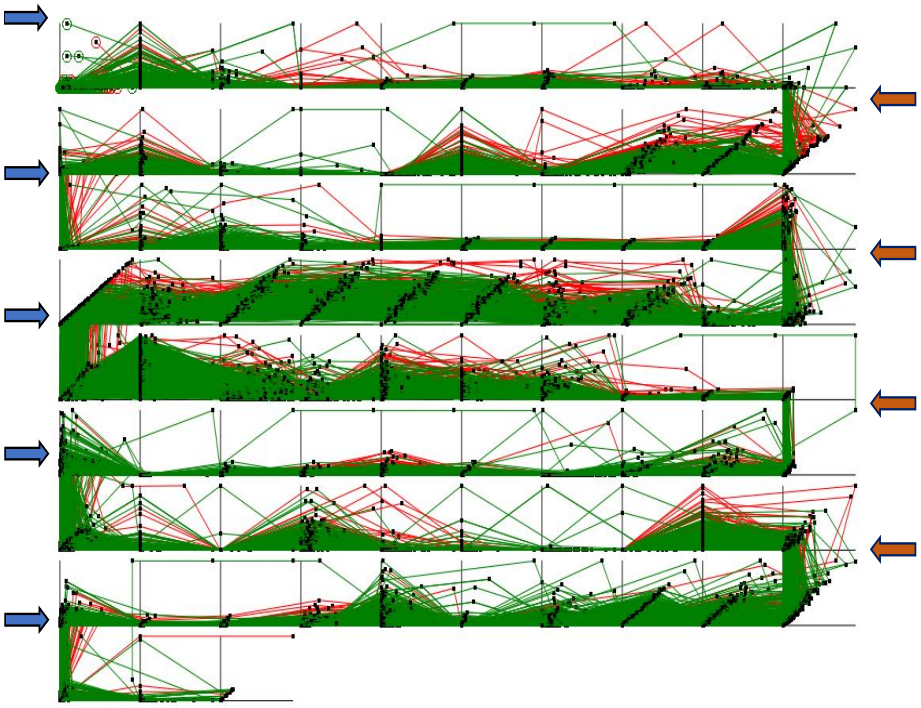
Interactive controls like non-linear scaling, non-orthogonal coordinates etc. allow improving visual discrimination of classes. However, using interactive visualization alone does not completely perform the data separation. It only provides a base for the class separation in terms of visual discrimination.

This chapter uses the IVLC algorithm that generates analytical rules to perform further class separation after reordering the coordinates. This algorithm generates these rules mainly using the threshold values generated from non-linear scaling. The rules belong to the class of rules proposed in [12].

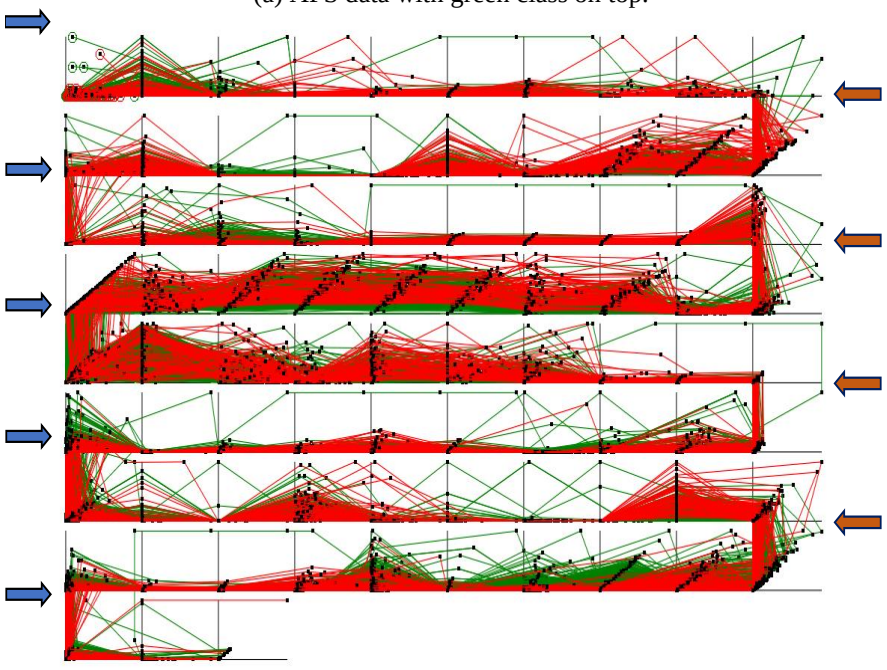
Visualizing data with larger dimensions become challenging in SPC. To display data of larger dimensions, a modified version of the SPC called as **Serpent Coordinate System** (**SCS**) is proposed. It is visualized in a grid like structure to accommodate all the dimensions on a single screen. Air Pressure System (APS) failure at Scania trucks [5] consists of 2 classes and 170 dimensions wherein 4 dimensions we removed since all the data points under those columns were 0 and were not informative. The data with 166 dimensions are displayed in Figure 7 and the coordinate labels corresponding each coordinate pair are displayed in Table 1.

TABLE 1: Coordinate Labels for Serpent Coordinate System for Figures 7a and 7b.

(X_1, X_2)	(X_3, X_4)	(X_5, X_6)	….	(X_{15}, X_{16})	(X_{17}, X_{18})	(X_{19}, X_{20})
(X_{21}, X_{22})	(X_{23}, X_{24})	(X_{25}, X_{26})	….	(X_{35}, X_{36})	(X_{37}, X_{38})	(X_{39}, X_{40})
(X_{41}, X_{42})	(X_{43}, X_{44})	(X_{45}, X_{46})	….	(X_{55}, X_{56})	(X_{57}, X_{58})	(X_{59}, X_{60})
(X_{61}, X_{62})	(X_{63}, X_{64})	(X_{65}, X_{66})	….	(X_{75}, X_{76})	(X_{77}, X_{78})	(X_{79}, X_{80})
(X_{81}, X_{82})	(X_{83}, X_{84})	(X_{85}, X_{86})	….	(X_{95}, X_{96})	(X_{97}, X_{98})	(X_{99}, X_{100})
(X_{101}, X_{102})	(X_{103}, X_{104})	(X_{105}, X_{106})	….	(X_{115}, X_{116})	(X_{117}, X_{118})	(X_{119}, X_{120})
(X_{121}, X_{122})	(X_{123}, X_{124})	(X_{125}, X_{126})	….	(X_{135}, X_{136})	(X_{137}, X_{138})	(X_{139}, X_{140})
(X_{141}, X_{142})	(X_{143}, X_{144})	(X_{145}, X_{146})	….	(X_{155}, X_{156})	(X_{157}, X_{158})	(X_{149}, X_{160})
(X_{161}, X_{162})	(X_{163}, X_{164})	(X_{165}, X_{166})				



(a) APS data with green class on top.



(b) APS data with red class on top.

FIGURE 7: APS failure at Scania trucks data (166D) visualized in SCS.

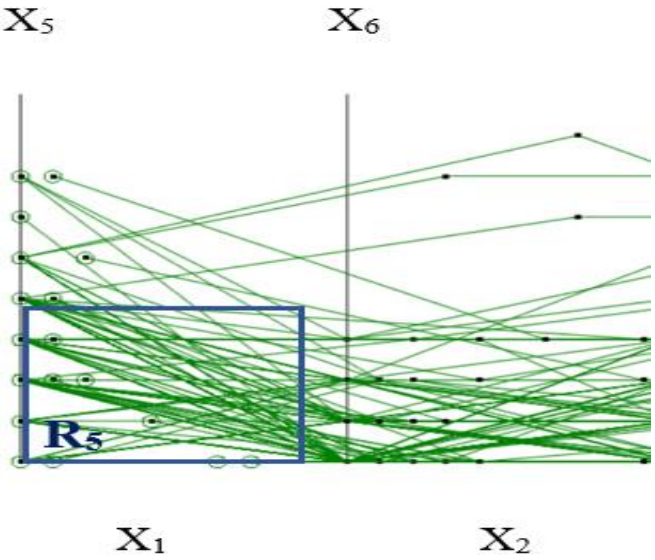
CHAPTER III

METHODS FOR INTERACTIVE DATA CLASSIFICATION

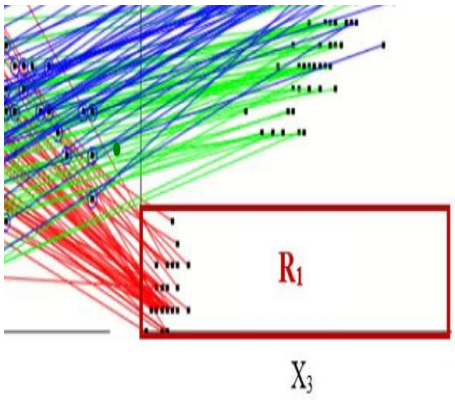
Iterative Visual Logical Classifier (IVLC) Algorithm

Below we present the *Iterative Visual Logical Classifier* (IVLC) algorithm that classifies data in iterations. It is different from other logical classifiers since it performs classification in visual space. As discussed in Chapter II, once we reorder the coordinates to find good vertical separation and obtain the vertical threshold values from non-linear scaling, the analytical rules are generated interactively based on these threshold values. This process of reordering and generating analytical rules are continued until we cover all the data in given dataset. The algorithm generates set of interpretable analytical rules for data classification. All steps can be conducted by the end user as a self-service. The output of IVLC algorithm results in series of rectangular areas. Different outputs of IVLC algorithm are displayed in Figure 8. The steps are listed below.

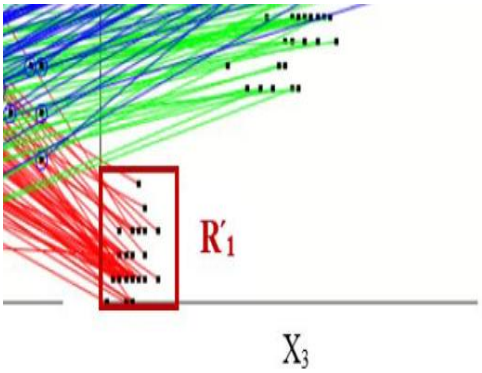
Step 1: Reorder the coordinates to find a good vertical separation for classes and perform non-linear scaling to get the threshold values along the vertical coordinates. Reordering of coordinates and non-linear scaling is performed interactively using SPCVis software system.



(a) Example of area (R_5) generated by IVLC for WBC data.



(b) Overgeneralized Area R_1 generation (larger part of the area is empty) for Iris (4D) data [5].



(c) Optimized Area R'_1 generation for Iris data.

FIGURE 8: Outputs of IVLC algorithm.

Step 2: Generate the analytical rules mainly based on the threshold values obtained from non-linear scaling from the previous step. For example, if we denote the set of areas generated as R_{class1} and R_{class2}, then the classification rules for n-D point $\mathbf{x} = (x_1, x_2, x_3, \ldots, x_n)$ are defined in Equations 3.1 and 3.2.

$$\text{If } x_i \in R_{class1}, \text{ then } \mathbf{x} \in \text{ class 1} \tag{3.1}$$

$$\text{If } x_j \in R_{class2}, \text{ then } \mathbf{x} \in \text{ class 2} \tag{3.2}$$

Step 3: The data that do not follow the rules generated in step 2 are used as input for next step. Also, in this step, the analytical rules can be tuned to avoid overgeneralization [14]. Figure 8b displays the generation of R_1 area where a large part of the area is empty. The area can be reduced by generating the area R'_1 of smaller dimension to avoid overgeneralization (see Figure 8c).

Step 4: The final step is to repeat the steps above until all the data are covered.

Model Evaluation with Worst-case *k* fold Validation Approach

Although Cross Validation is a common technique used for model evaluation, it comes with its own challenges. Due to the random split of training and validation data, we might observe a bias in the estimated average error rate. Also, if we consider all the possible splits, it becomes computationally challenging to find all the combinations since the number of splits grows exponential with the number of given data points [13].

In order to overcome this challenge, we use a worst case heuristics technique to split the data into training and validation sets in *k*-fold cross validation. The worst case fold contains the data of one class that are similar to cases of the opposing class [13] making classification more challenging with higher number of misclassification than in the traditional random *k*-fold cross validation. If the algorithm produces high accuracy in the worst-case fold, then the average case accuracy produced by the traditional random *k*-fold cross validation is expected to be greater.

In this chapter, the worst-case fold is extracted using visual representation of the data in SPC system. As already mentioned, the data are displayed in such a way that they tend to be separated along the vertical axes in the SPC. For instance, if the dataset contains class *A* and class *B* with class *B* at the bottom and class *A* at the top in SPC visualization, then the worst-case validation split contains the cases with class *A* displayed on the bottom along with class *B* and vice versa. Since 10-fold cross validation is used in our classification model, first validation fold contains the top 10% of the worst-case data n-D points, next validation fold contains the next 10% of the worst-case data and so on.

Experiments with Interactive Data Classification Approach

Iris Data (4D)

The first dataset is the Iris data [5]. It has 4 dimensions (sepal length, petal length, sepal width and petal width) with a total of 150 cases. The data consists of three classes

namely setosa, versicolor and virginica, each class consisting of 50 cases. Figure 9 displays the data in SPCVis software system. The four dimensions are denoted as X_1, X_2, X_3 and X_4 coordinates. Class 1 separation is defined by the rule in Equation 3.3.

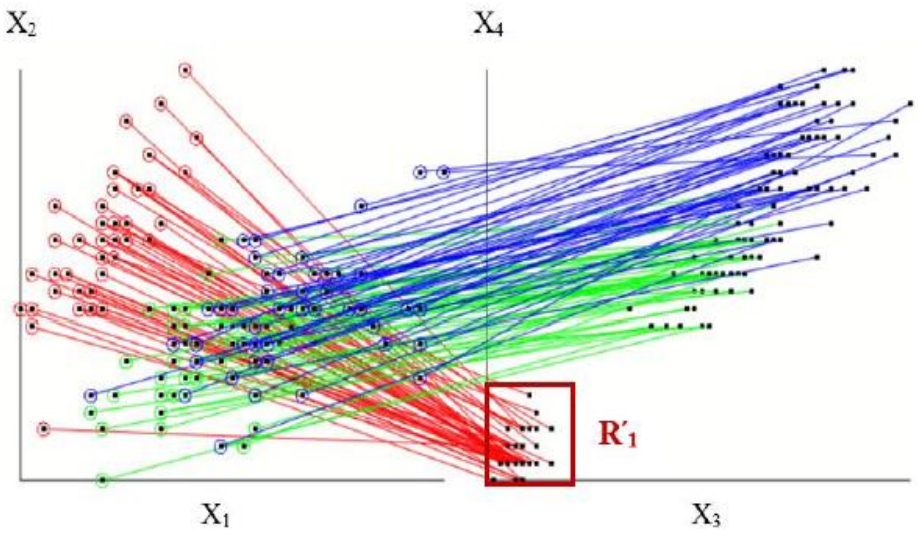


FIGURE 9: Visualization of rule for R'_1 on Iris data for class 1 separation.

$$\text{If } (x_4, x_3) \in R'_1, \text{ then } \mathbf{x} \in \text{class 1.} \tag{3.3}$$

The optimized coordinate order for separation of classes 1 and 3 are (X_1, X_3) and (X_2, X_4) with X_3 and X_4 as vertical coordinates. Separation criteria for class 2 and class 3 is given in Equation 3.4.

$$\text{If } (x_1, x_2, x_3, x_4) \in R_2 \text{ then } \mathbf{x} \in \text{class 2, else } \mathbf{x} \in \text{class 3.} \tag{3.4}$$

We can further refine rule defined by $R'_2 = R_{21}$ & R_{22} by adding another area R_3 to form a new optimized rule in Equation 3.5.

$$\text{If } (x_1, x_2, x_3, x_4) \in R'_2 \text{ or } R_3, \text{ then } \mathbf{x} \in \text{class 2, else } \mathbf{x} \in \text{class 3.} \quad (3.5)$$

Figure 10 visualizes the above rule for separation classes 2 and 3. The accuracy obtained in 10-fold cross validation with worst case split is **100%.**

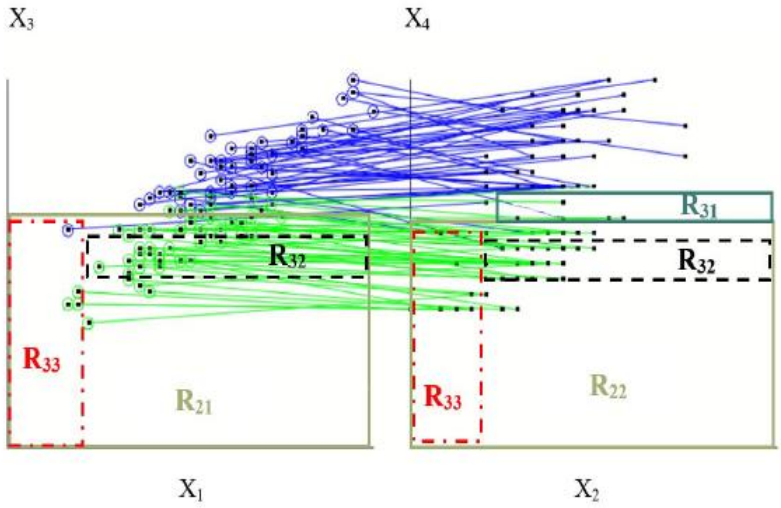


FIGURE 10: Visualization of rule for R_2 and R_3 on Iris data for classes 2 and 3 separation.

Wisconsin Breast Cancer (WBC) Data (9D)

The second dataset is Wisconsin Breast Cancer (WBC) dataset [5]. It contains 699 cases of data with 9 features. In this dataset, 16 cases were incomplete and hence were removed. Remaining data with 683 cases consists of 444 benign cases and 239 malignant

instances. Figure 3 displays WBC data after loading in SPCVis software system. Figure 11 visualizes analytical rules for R_5 and R_6 generated for benign class classification.

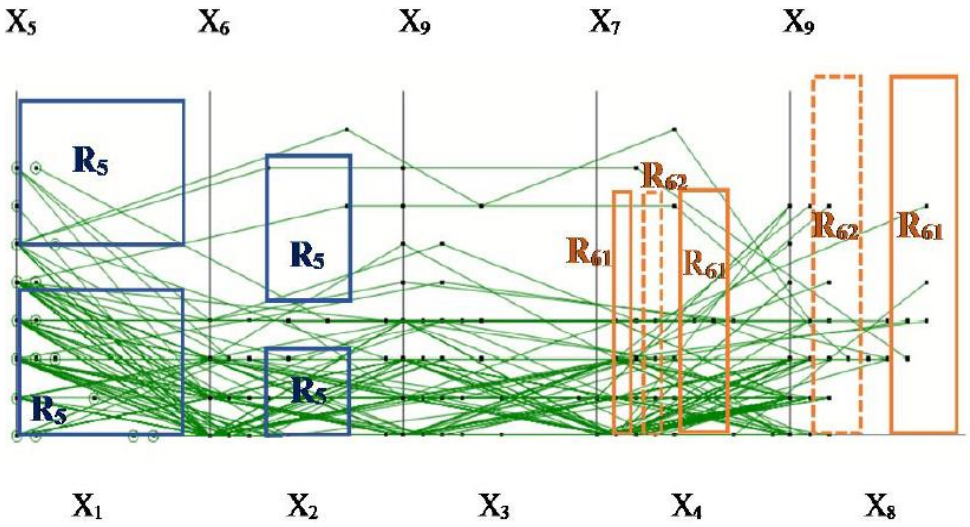


FIGURE 11: Visualization of rules for R_5 and R_6 on WBC data.

The analytical rules for R_5 and R_6 for classification of class 1 is defined in (3.6).

$$\text{If } (x_1, x_5, x_2, x_6, x_4, x_7, x_8, x_9) \in R_5 \,\&\, R_6, \text{ then } x \in \text{class } 1 \qquad (3.6)$$

$$R_6 = R_{61} \text{ or } R_{62}. \qquad (3.7)$$

The accuracy obtained after 10-fold cross validation technique with worst case heuristics is **99.56%.**

Seeds Data (7D)

The third dataset consists of seeds data with 7 dimensions and 210 instances. The data contain three classes: Kama, Rosa and Canadian, based on the characteristics of wheat kernels like seed perimeter area, length and width of kernel etc. Each class consists of 70 instances [5]. The data are loaded, and the coordinates are reordered to find the

prominent class separation along vertical coordinates. The analytical rules are generated based on the vertical separation. Figure 12 displays seeds data with areas for analytical rules R_1 and R_2 for class 2 (green) separation. Due to odd number of coordinates X_2 coordinate is duplicated as the 8^{th} coordinate since X_2 coordinate provided good visual discrimination between the classes. The rule for class 2 classification is defined in (3.8). The accuracy obtained after 10-fold cross validation technique with worst case heuristics is **100%.**

$$\text{If } (x_1, x_7, x_4, x_2, x_6) \in (R_1 \text{ or } R_2), \text{ then } \mathbf{x} \in \text{class 2} \tag{3.8}$$

$$R_2 = R_{21} \text{ or } R_{22} \tag{3.9}$$

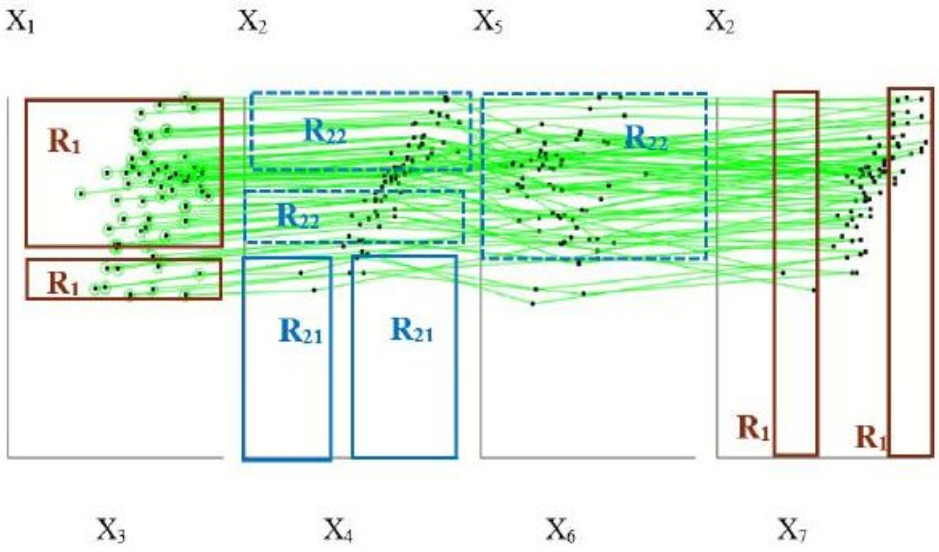


FIGURE 12: Visualization of rules for R_1 and R_2 on Seeds data (7D) for class 1 separation with all the cases from class 2.

CHAPTER IV

METHODS FOR AUTOMATED DATA CLASSIFICATION

In most of the scenarios, interactive approach to data classification works well for small dataset. When handling large data, this approach becomes tedious and time consuming. Also, due to the display of large number of data instances within limited space would lead to occlusion and becomes challenging for the end use to find the patterns. To overcome this problem, automation is used where the patterns are discovered automatically by the machine with minimal human intervention. Automation is implemented using two algorithms below.

- Coordinate Order Optimizer (COO) algorithm
- Genetic Algorithm (GA)

These algorithms are used in combination with non-linear scaling, to enhance the data interpretability. Figure 13 provides the overview of the automated data classification approach.

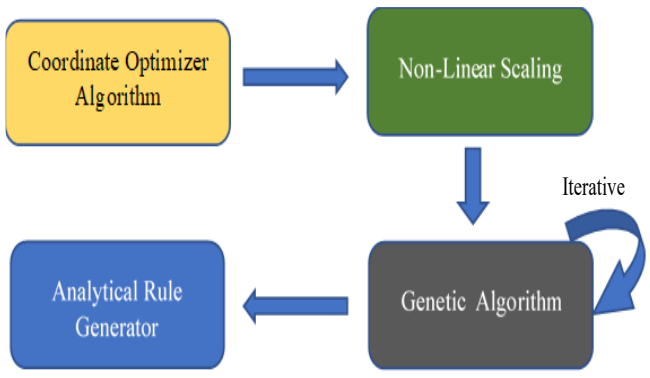


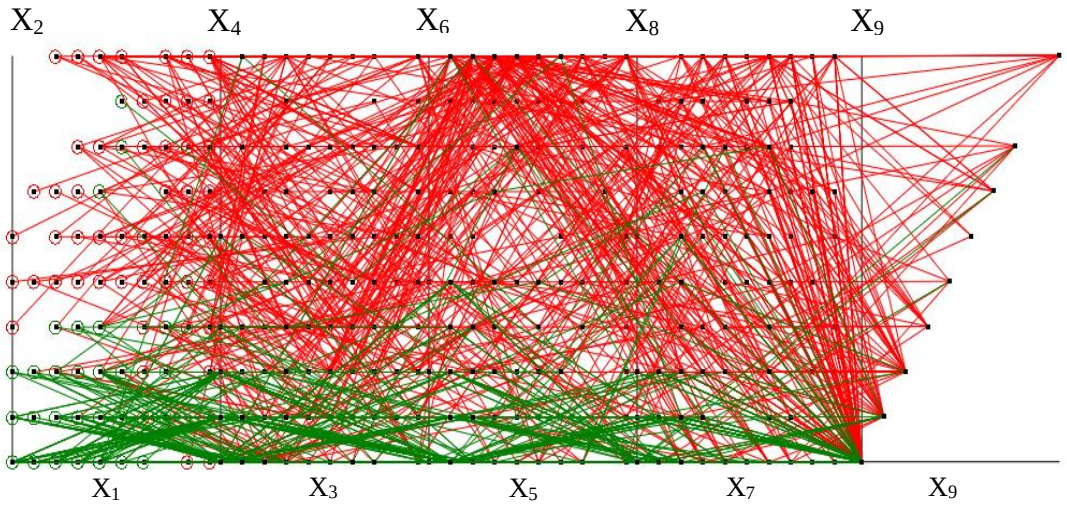
FIGURE 13: Overview of automation for data classification in SPCVis.

Coordinate Order Optimizer (COO) Algorithm

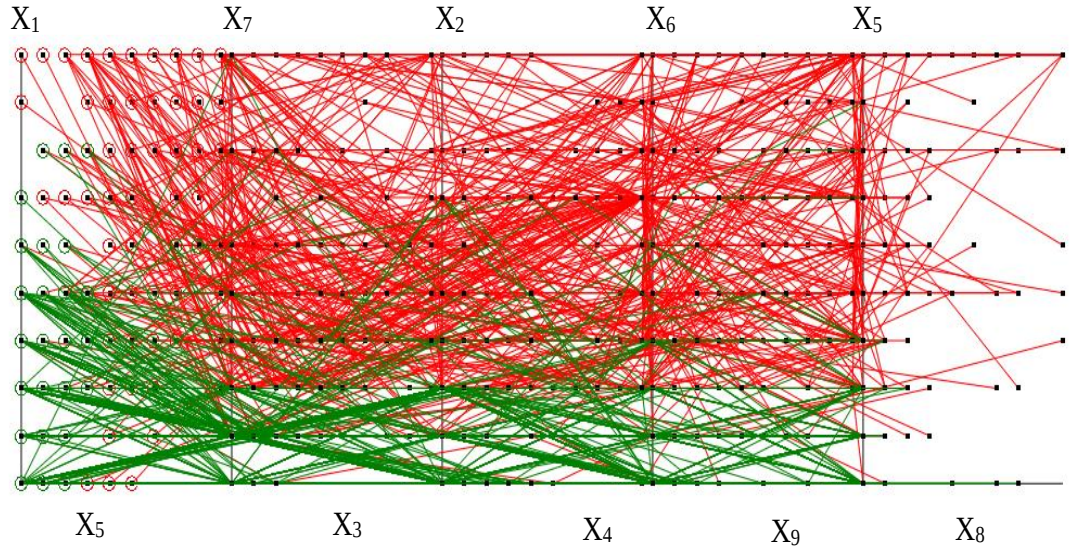
COO algorithm optimizes the order of coordinates by primarily using Coefficient of Variation (CV) [7] parameter, also called as relative standard deviation (RSD) defined as the ratio of the standard deviation σ to the mean μ of the given data sample.

$$C_v = \frac{\sigma}{\mu} \tag{4.1}$$

CV is a standardized measure of dispersion of data distribution. It is computed individually per coordinate for each class. Next, the mean of all CVs of classes of each coordinate is calculated. Lesser the mean of CV, lesser the data dispersion along the coordinate. The coordinate with the least mean of CV is considered the *best coordinate*. Hence, the coordinates are arranged in the descending order of the mean CV values. Figures 14a and 14b display Breast Cancer data before and after order of coordinates is optimized, respectively. In the optimized order of coordinates (Figure 14b), the green class is settled at the bottom and red class on the top whereas in the Figure 14a, more green lines are at the top along with red class and more red lines in the bottom along with green lines.



(a) WBC data before optimization of order of coordinates.



(b) WBC data after optimization of order of coordinates (lesser green cases on top).

FIGURE 14: Visualization of WBC data before and after applying COO algorithm.

Non-Linear Scaling: The threshold for all the vertical coordinates is calculated from the average of the bottom class taken from vertical coordinates. Non-linear scaling is performed using Equation 2.1 for all the vertical coordinates. This improves data

interpretability and provides better visualization of separation of classes. The output of non-linear scaling is displayed in Figure 4 that enhances the visual separation of classes.

Genetic Algorithm (GA)

GA is used to generate the optimized areas of high fitness or purity [4] based on which the analytical rules are created for further classification. An overview of the implementation of GA in our approach is shown in Figure 15 for discovering the areas for classification. In this context, areas are referred as Area of Interest (AOI).

Initial Population Selection: The initial population contains randomly generated rectangles (see Figure 16) defined by the fixed ratio *r* of each coordinate. For instance, *r* can be 0.1 of length of the coordinate. The data are normalized to [0,1] interval. The generation of the AOI in the SPCVis using GA is an iterative process where each iteration creates a generation [1] of a new set of AOIs. Before generating the AOIs, a search space [2] is defined containing maximum cases belonging to the class on which we build analytical rules. Consider a situation where analytical rules are being built to classify class *k*. Let $x_{imax}(k)$ and $x_{imin}(k)$ be the maximum and minimum data points respectively belonging to coordinate X_i and $x_{jmax}(k)$ and $x_{jmin}(k)$ be the maximum and minimum data points respectively belonging to coordinate X_j. The number of Areas of Interests N_{AOI} generated to classify class *k* within a coordinate pair (X_i, X_j) is given in Equations 4.2 and 4.3.

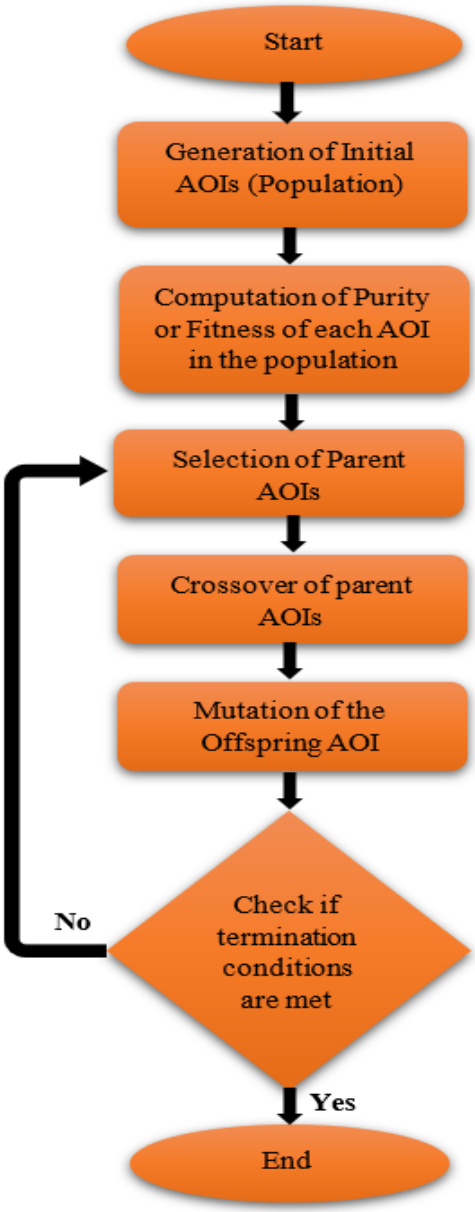


FIGURE 15: GA flow chart used in SPCVis data classification.

$$N_{AOI} = \frac{S(k)}{r^2} \tag{4.2}$$

$$S(k) = (x_{imax}(k) - x_{imin}(k))\,(x_{jmax}(k) - x_{imin}(k)) \tag{4.3}$$

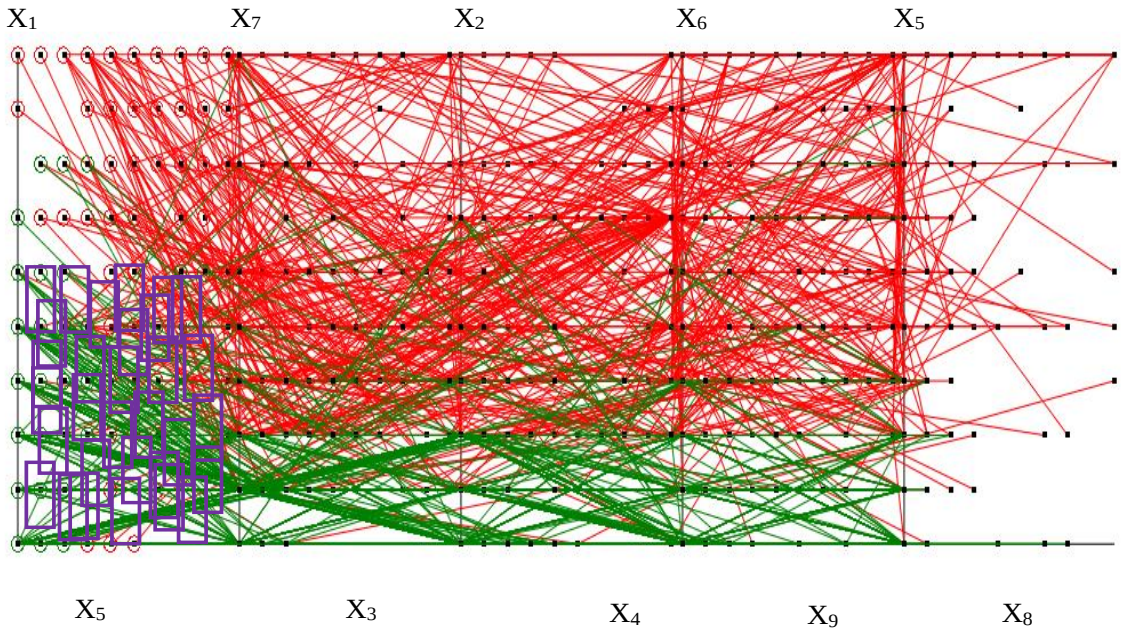


FIGURE 16: Random generation of areas in WBC data.

Parents Selection: This stage involves selection of the AOIs or parents to generate an AOI called the offspring, i.e., combining two areas to form a bigger area. The parents are selected based on two criteria (1) *Purity* or fitness, and (2) *Proximity*.

Purity or fitness of an AOI with respect to class *k* for a given pair of coordinates (X_i, X_j) is defined as the ratio of the number of data points belonging to class *k* to the total number of data points within the AOI. Let AOI_t be an AOI in the coordinate pair (X_i, X_j). The purity $P_k(AOI_t)$ of a single AOI_t with respect to class *k* is given in Equation (4.4). $N_k(AOI_t)$ is the number of points (x_i, x_j) in AOI_t in (X_i, X_j) that belong to lines from class C_k and is defined in Equation 4.5. $N(AOI_t)$ is the total number of points (x_i, x_j) within a given AOI_t in (X_i, X_j) and is defined in Equation 4.6.

$$P_k(AOI_t) = \frac{N_k(AOI_t)}{N(AOI_t)} \quad (4.4)$$

$$N_k(AOI_t) = ||\{(x_i, x_j): (x_i, x_j) \in AOI_t \& C_k\}|| \quad (4.5)$$

$$N(AOI_t) = ||\{(x_i, x_j): (x_i, x_j) \in AOI_t\}|| \quad (4.6)$$

After computing the purity of the AOIs, the parents with closest *Proximity* (nearest parents) are selected for the next stage, i.e., the crossover. While *Proximity* can be defined in multiple ways, in our experiments we applied the Euclidean distance between the mid-point of two areas within the given coordinate pair commonly used in GA for similar tasks [2]. For instance, given the mid-point of two AOIs $\{(x_{im1}, x_{jm1}), (x_{im2}, x_{jm2})\}$ within coordinate pair (X_i, X_j), the *Proximity* of two AOIs is given in Equation 4.7.

$$Proximity = \sqrt{(x_{im1} - x_{im2})^2 + (x_{jm1} - x_{jm2})^2} \quad (4.7)$$

Crossover: Once the parents with highest *Purity* and closest *Proximity* (nearness) are selected, the parent AOIs are combined to form a new AOI (offspring). A single parent AOI_{Pg} is represented in Equation 4.8. Here x_1, x_2, y_1, y_2 are left, right, bottom ant top coordinates of the rectangular AOI_{Pg} and is represented in the Figure 17.

$$AOI_{Pg} = P_g(x_1, x_2, y_1, y_2) \quad (4.8)$$

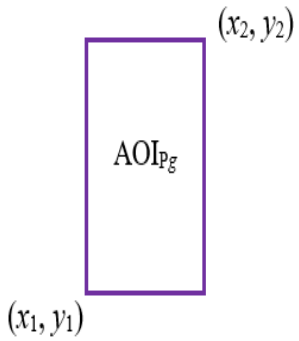


FIGURE 17: Representation of single parent AOI_{Pg} from generation g.

A crossover of two parents AOI_{P1g} and AOI_{P2g} from generation g to produce an offspring AOI_{O1g} within a given coordinate pair (X_i, X_j) is represented in Equation 4.9. The two parents are defined in Equations 4.10 and 4.11. The function $F(AOI_{P1g}, AOI_{P2g})$ is defined in Equation 4.12 as an envelope around these two AOIs. Figure 18 displays different types of crossovers of the parent AOIs (with and without overlapping, or diagonally overlapping).

$$AOI_{O1g} = F(AOI_{P1g}, AOI_{P2g}) \quad (4.9)$$

$$AOI_{P1g} = P_{1g}(x_{11}, x_{12}, y_{11}, y_{12}) \quad (4.10)$$

$$AOI_{P2g} = P_{2g}(x_{21}, x_{22}, y_{21}, y_{22} \quad (4.11)$$

$$F(AOI_{P1g}, AOI_{P2g}) = \{\min(x_{11}, x_{21}), \max(x_{12}, x_{22}), \min(y_{11}, y_{21}), \max(y_{12}, y_{22})\} \quad (4.12)$$

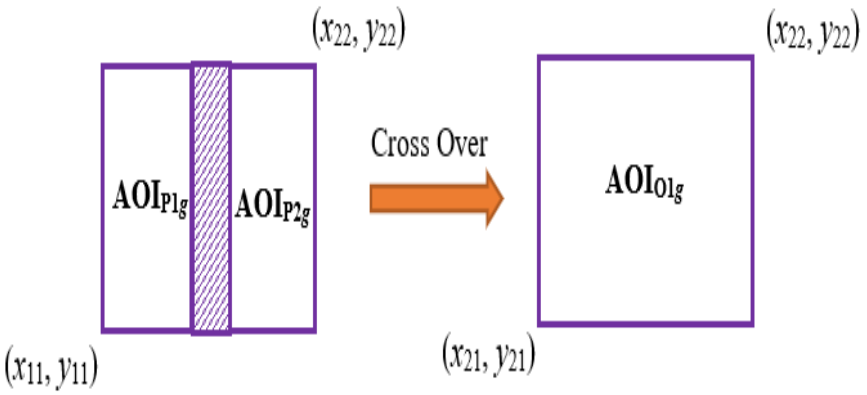
Mutation: In GA, certain characteristics of the offspring generated from the previous generation are modified (mutated) in order to speed up the process of reaching an optimized solution. This includes modifying the characteristics of the offspring either

by flipping, swapping or shuffling the properties that represent the offspring. For instance, if the offspring is represented by bits, then the mutation by flipping would include switching some of the bits from 0 to 1 or vice versa [18].

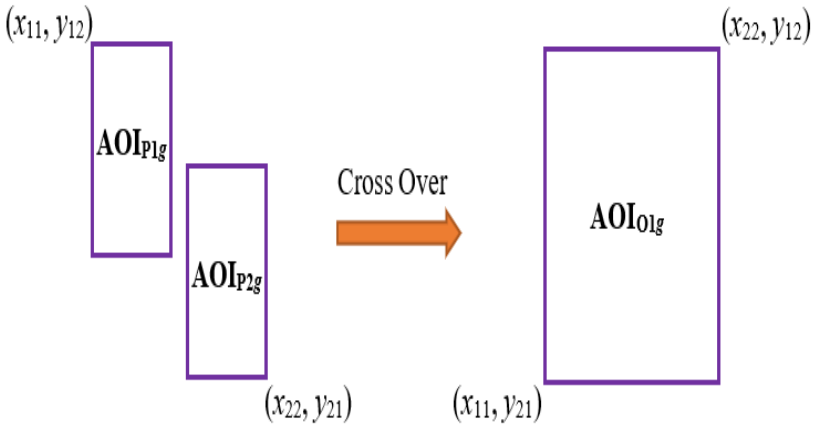
Since the objective of mutation is to generate an offspring with better characteristics than its parents, in our proposed technique, we generate the mutated offspring by interactively *generating a new parent AOI* with high *Purity* and close *Proximity* with the automatically generated parent AOI. This results in an offspring with better characteristics compared to its parents in terms of size and *Purity*. Figure 19a represents the automatically generated parent AOI in purple (straight line) and interactively generated parent AOI (dotted lines). The resulting mutated offspring AOI has superior characteristics compared to its parents with high *Purity* and larger area compared to its previous generation as displayed in Figure 19b.

Termination: Since GA process is iterative, there are several conditions based on which the process can be terminated. In our proposed method, we use two techniques as the termination criteria: (1) areas with highest *Purity* or fitness (100%) are generated, or (2) manual inspection termination in SPCVis. If either of the two criteria is met, the process is terminated.

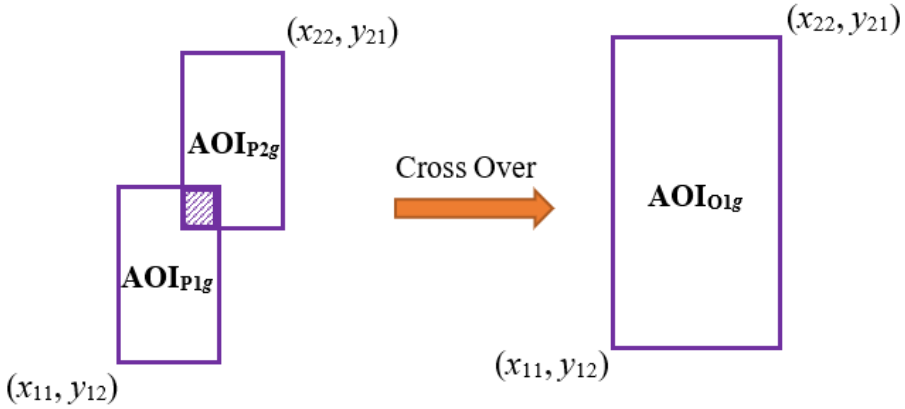
Analytical Rule Generator: This is similar to the second step performed in IVLC algorithm as discussed in Chapter III. The only change that the areas used here are generated from GA whereas in Chapter III, the areas used are interactively generated.



(a) Crossover of two overlapping parent AOIs.

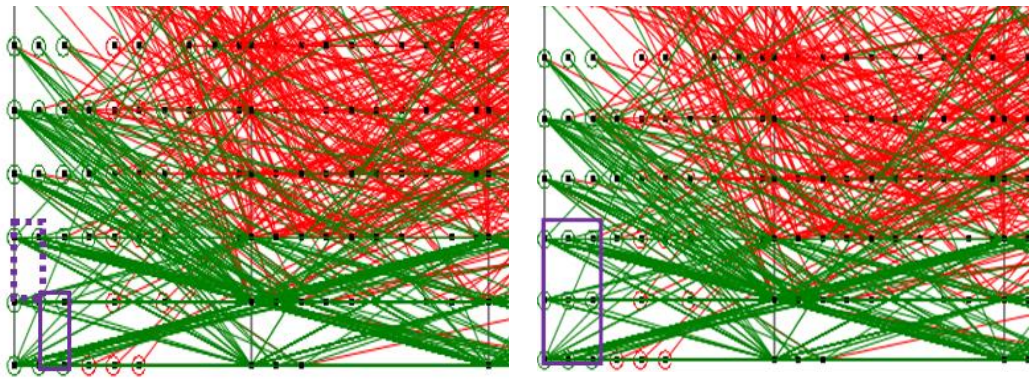


(b) Crossover of two non - overlapping parent AOIs.



(c) Crossover of two diagonally overlapping parent AOIs.

FIGURE 18: Different types of crossovers of two parent AOIs to generate offspring AOI.



(a) Parent AOI (dotted lines) generated automatically (straight lines) and interactively (dotted lines) with high purity in generation *g*.

(b) Mutated Offspring in generation $g + 1$.

FIGURE 19: Visualizations of consecutive generations of AOIs in WBC data.

Experiments with Automated Data Classification Approach

Experiments are conducted with same datasets mentioned in Chapter III i.e., WBC, Iris and Seeds datasets. In addition to these datasets, experiment is also conducted on Air Pressure System (APS) Failure at Scania trucks. This dataset consists of 60,000 instances with 170 features. Compared to interactive approach, the automated techniques provided better results with smaller number of areas and less iterations.

Iris Data (4D)

Class 1 of Iris data is classified by running the COO algorithm and GA. The optimized order of the coordinates for class 1 classification is (X_4, X_3) and (X_1, X_2). It is displayed in Figure 20. The rule for class 1 (green) separation defined in Equation 4.13.

$$\mathbf{r}_1\text{: If } (x_4, x_3) \in R_{11}\text{, then } \mathbf{x} \in \text{ class 1} \tag{4.13}$$

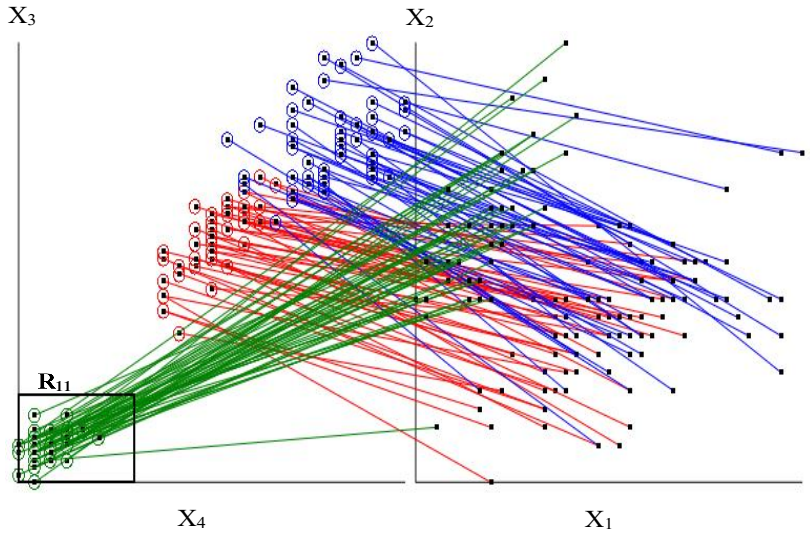


FIGURE 20: Visualization of Iris data with class 1 separation rule.

After class 1 separation, COO is run again for classifying the remaining classes. The optimized order of coordinates for class 2 and class 3 separation are (X_1, X_3) and (X_2, X_4). The visualization of classes 2 and 3 after reordering the coordinates is shown in Figure 21.

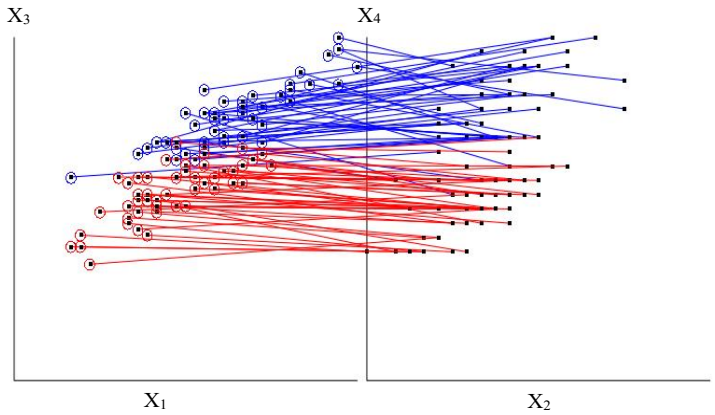


FIGURE 21: Visualization of Iris data with classes 2 and 3 after reordering the coordinates.

Figure 21 clearly displays separation of classes 2 and 3 along the vertical coordinates. This visualization is further enhanced by applying the non-linear scaling with following thresholds on coordinates: 0.7 on X_3 and 0.71 on X_4.

GA is run on class 2 and class 3 data to generate the areas. Visualization of non-linear scaling along with the areas are displayed in Figure 22a with 10 cases. Figure 22b displays the same visualization with all the cases from class 2 and class 3.

The areas R_2 and R_3 for classes 2 and 3 separation are defined in Equations 4.14 and 4.16 and their corresponding rules are defined in Equations 4.15 and 4.17.

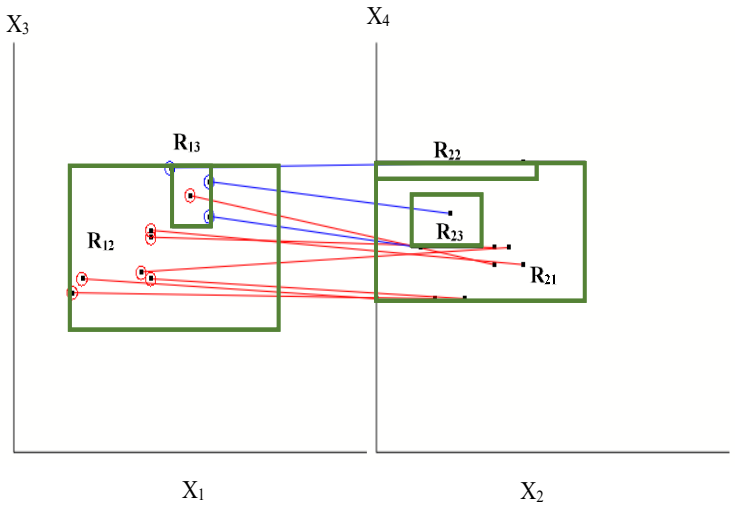
$$R_2 = R_{12} \,\&\, R_{21} \,\&\, (\neg R_{13} \,\&\, \neg R_{23}) \,\&\, \neg R_{22} \tag{4.14}$$

$$\mathbf{r_2}\text{: If } (x_1, x_2, x_3, x_4) \in R_2 \text{ , then } \mathbf{x} \in \text{ class 2} \tag{4.15}$$

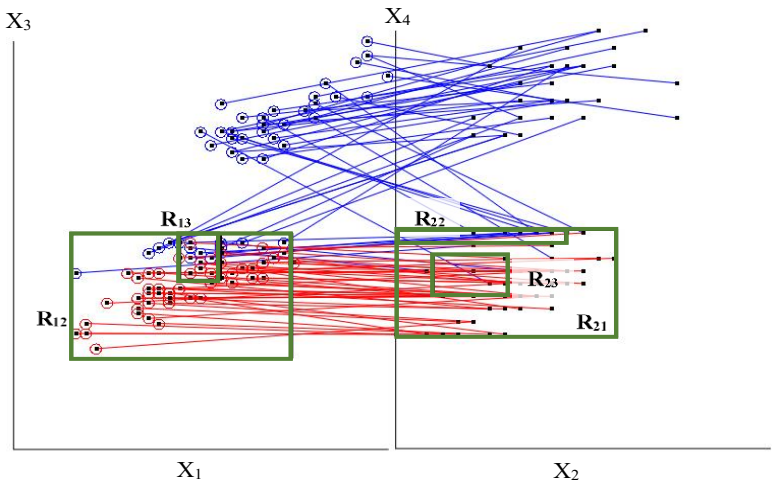
$$R_3 = (\neg R_{11} \,\&\, \neg R_2) \tag{4.16}$$

$$\mathbf{r_3}\text{: If } (x_1, x_2, x_3, x_4) \in R_3 \text{ , then } \mathbf{x} \in \text{ class 3} \tag{4.17}$$

The area parameters generated for Iris data classification is listed in Table 2. The accuracy obtained for Iris data classification with 10-fold cross validation using worst-case heuristics approach is **100%**.



(a) Visualization of rule $\mathbf{r_2}$ on Iris dataset for classes 2 and 3 separation with 10 cases.



(b) Visualization of rule $\mathbf{r_2}$ on Iris dataset for classes 2 and 3 separation with all the cases.

FIGURE 22: Visualization of rule $\mathbf{r_2}$ on Iris dataset for classes 2 and 3 separation.

Table 2. Parameters of the areas generated for Iris data classification.

	Rectangle Parameters				
	Left	Right	Bottom	Top	Coordinate Pair
R_{11}	0.0	0.3	0.0	0.2	(X_1, X_3)
R_{12}	0.16	0.75	0.3	0.7	(X_1, X_3)
R_{13}	0.45	0.56	0.55	0.7	(X_1, X_3)
R_{21}	0.0	0.59	0.37	0.71	(X_2, X_4)
R_{22}	0.0	0.45	0.67	0.71	(X_2, X_4)
R_{23}	0.1	0.3	0.5	0.63	(X_2, X_4)

Wisconsin Breast Cancer (WBC) Data (9D)

The second dataset is WBC dataset, as discussed in Chapter III. Running the COO algorithm produced the following order of coordinates: (X_5, X_1), (X_3, X_7), (X_4, X_2), (X_9, X_6) and (X_8, X_5). Figure 23a displays WBC data visualized in SPCVis with 12 cases of class 1 data along with the areas.

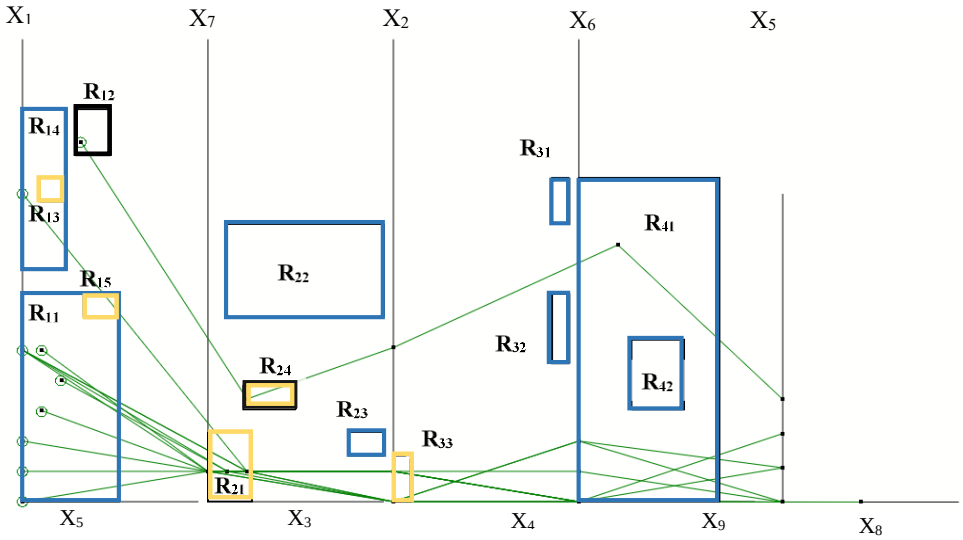
The rectangle R_{km} is m^{th} rectangle in the k^{th} pair of coordinates. For instance, in Figure 23a, rectangle R_{24} is a 4^{th} rectangle in the second pair of coordinates that is (X_3,X_7). Figure 23b displays all the cases from both classes of WBC data along with the non-linear scaling with following thresholds on coordinates: 0.6 on X_1, 0.25 on X_7 and X_2 and 0.3 on X_6. The areas R_1-R_3 are defined in Equations 4.18 – 4.20. The rule **r** is defined in Equation 4.21. The area coordinates are given in the Table 3. The accuracy obtained after 10-fold cross validation technique with worst case heuristics is **99.71%.**

$$R_1 = R_{11} \& \neg R_{15} \ \& R_{41} \& \neg R_{42} \& (\neg R_{22} \text{ or } \neg R_{23}) \text{ or } (\neg R_{31} \text{ or } \neg R_{32}) \tag{4.18}$$

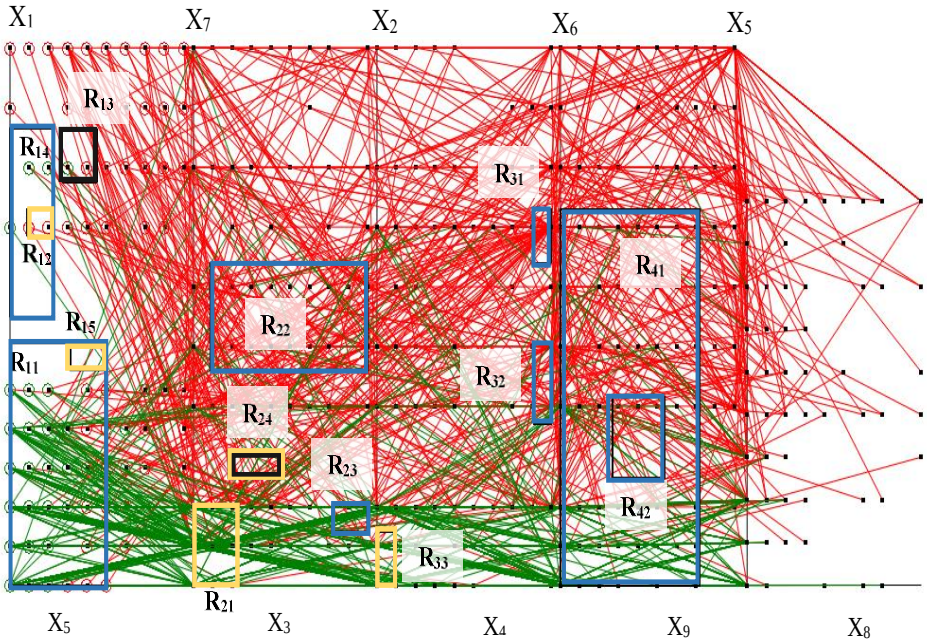
$$R_2 = R_{14} \& \neg R_{12} \ \& (\neg R_{21} \text{ or } \neg R_{24}) \text{ or } \neg R_{33} \tag{4.19}$$

$$R_3 = R_{13} \& R_{24} \tag{4.20}$$

$$\textbf{r}\text{: If } (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_9) \in R_1 \text{ or } R_2 \text{ or } R_3 \text{ then } \mathbf{x} \in \text{Class 1, else } \mathbf{x} \in \text{Class 2} \tag{4.21}$$



(a) Visualization of rule **r** on WBC dataset with 12 cases.



(b) Visualization of rule r on WBC dataset with all the cases.

FIGURE 23: Visualization of rule **r** on WBC dataset for class 1 separation.

TABLE 3. Parameters of the areas generated for WBC data classification.

	Rectangle parameters				
	Left	Right	Bottom	Top	Coordinate Pair
R_{11}	0.0	0.55	0.0	0.45	(X_5, X_1)
R_{12}	0.3	0.5	0.75	0.85	(X_5, X_1)
R_{13}	0.1	0.25	0.65	0.7	(X_5, X_1)
R_{14}	0.0	0.25	0.5	0.85	(X_5, X_1)
R_{15}	0.40	0.55	0.40	0.45	(X_5, X_1)
R_{21}	0.0	0.25	0.0	0.15	(X_3, X_7)
R_{22}	0.1	1.0	0.4	0.6	(X_3, X_7)
R_{23}	0.8	1.0	0.1	0.15	(X_3, X_7)
R_{24}	0.2	0.5	0.2	0.25	(X_3, X_7)
R_{31}	0.9	1.0	0.6	0.7	(X_4, X_2)
R_{32}	0.9	1.0	0.3	0.45	(X_4, X_2)
R_{33}	0.0	0.1	0.0	0.12	(X_4, X_2)
R_{41}	0.0	0.8	0.0	0.7	(X_9, X_6)
R_{42}	0.3	0.6	0.2	0.35	(X_9, X_6)

Seeds Data (7D)

Seeds data, as discussed in Chapter III consists of 7 dimensions. Due to odd number of dimensions, X_7 is duplicated to display the data in SPCVis (see Figure 24).

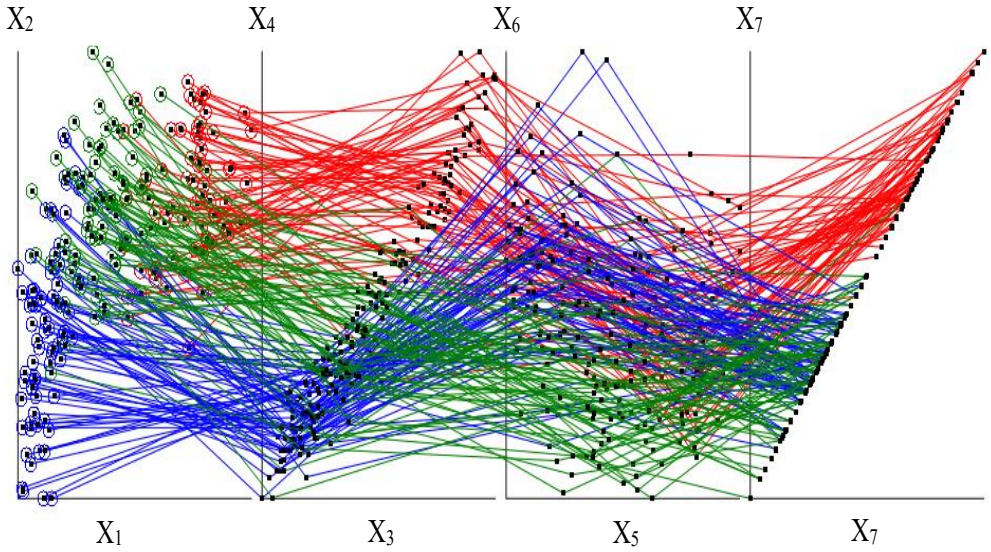


FIGURE 24: Visualization of Seeds data with all the three classes in SPCVis.

Analytical rules for class 3 (blue) are generated after running the COO algorithm and GA. The optimized order of the coordinates for class 1 classification is (X_3, X_1), (X_7, X_5), (X_6, X_2), and (X_3, X_4). Figure 25 represents the optimized order of coordinates.

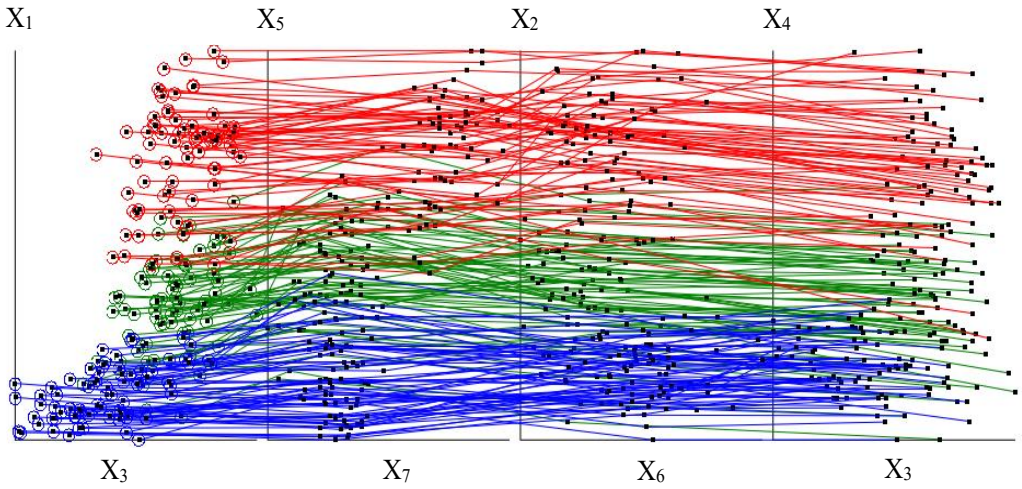


FIGURE 25: Visualization of Seeds data with all the three classes in SPCVis after coordinate order optimization.

Non-linear scaling is then performed with following thresholds on coordinates: 0.4 on X_1, 0.45 on X_5, 0.4 on X_2 and 0.4 on X_4. The visualization of classes 2 and 3 after reordering the coordinates and non-linear scaling is shown in Figure 26.

Areas are generated by running GA and analytical rules are built using these generated areas. The visualization of all the three classes with non-linear scaling and the areas for class 3 (blue) separation is displayed in Figure 27a. The area is defined in Equation 4.22. The parameters of the areas generated by GA are listed in Table 4.

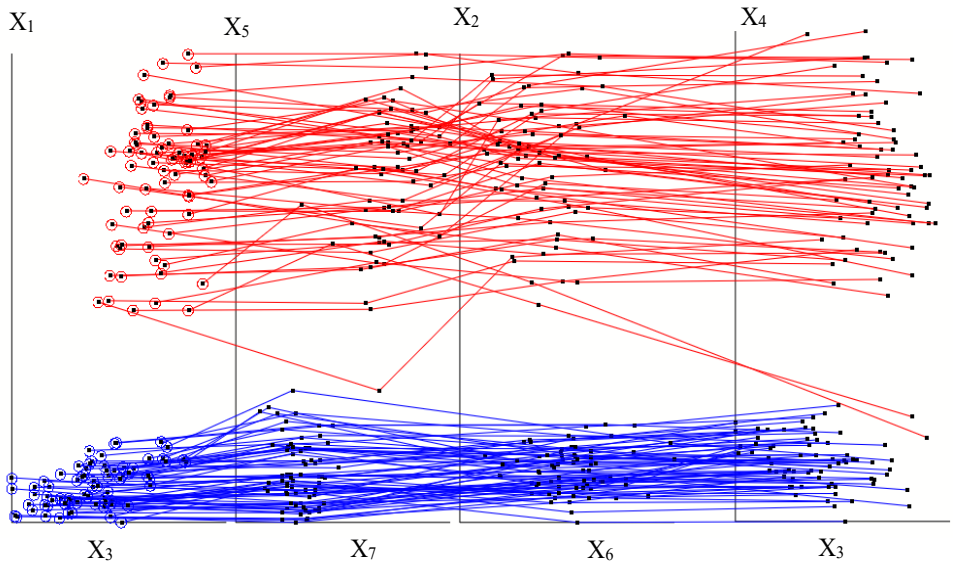


FIGURE 26: Visualization of Seeds dataset with classes 2 (red) and 3 (blue) after performing non-linear scaling on optimized order of coordinates.

$$R_1 = R_{11} \,\&\, R_{21} \,\&\, R_{41} \,\&\, (\neg R_{42} \text{ or } \neg R_{31}) \,\&\, (\neg R_{12} \,\&\, \neg R_{22} \,\&\, (\neg R_{32} \text{ or } \neg R_{43})) \qquad (4.22)$$

The rule $\mathbf{r_1}$ for class 3 classification is given in Equation 4.23.

$$\mathbf{r_1}\text{: If } (x_1, x_2, x_3, x_4, x_5, x_6, x_7) \in R_1, \text{ then } \mathbf{x} \in \text{ class 3} \qquad (4.23)$$

TABLE 4. Parameters of the areas generated for classification in Seeds data in 1st iteration.

	Rectangle Parameters				
	Left	Right	Bottom	Top	Coordinate Pair
R_{11}	0.0	0.85	0.0	0.3	(X_3, X_1)
R_{12}	0.5	0.85	0.12	0.26	(X_3, X_1)
R_{21}	0.1	0.55	0.0	0.43	(X_7, X_5)
R_{22}	0.27	0.55	0.2	0.43	(X_7, X_5)
R_{31}	0.58	0.72	0.25	0.35	(X_6, X_2)
R_{32}	0.0	0.45	0.21	0.3	(X_6, X_2)
R_{41}	0.32	0.8	0.25	0.36	(X_3, X_4)
R_{42}	0.32	0.4	0.25	0.3	(X_3, X_4)
R_{43}	0.45	0.55	0.0	0.15	(X_3, X_4)

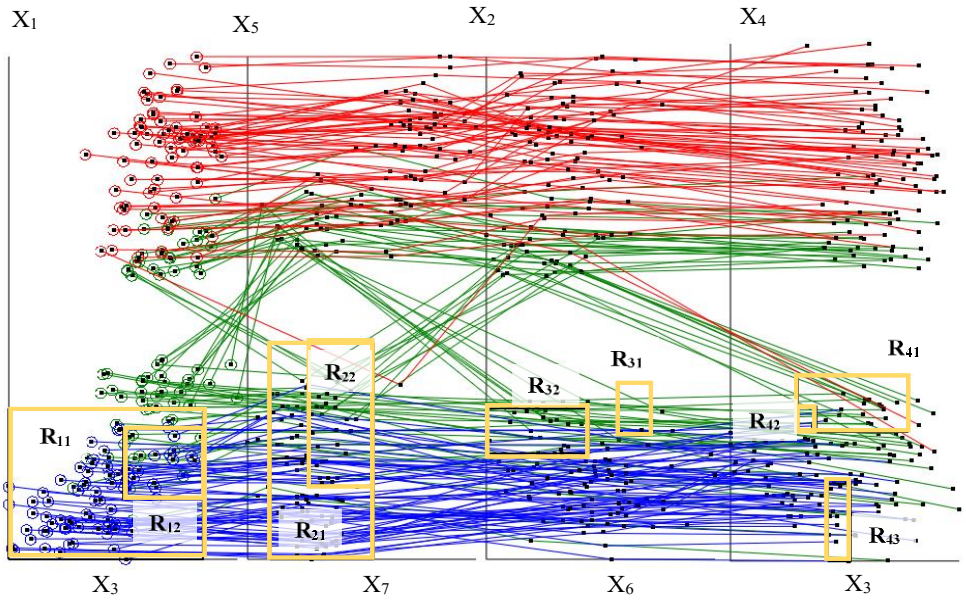
The cases that do not follow the rule generated for class 3 is sent to the next iteration where rules are generated for other classes. Areas are generated again by running GA and analytical rules are built using these generated areas. The visualization with all the three cases along with non-linear scaling and the areas are displayed in Figure 27b. In this case the rule is generated for class 2 (red). The optimized order of the coordinates remains the same. Non- linear scaling with the same threshold as in the previous iteration is performed on the vertical coordinates. The areas R_2 and R_3 are defined in Equations 4.24 and 4.25. The rules $\mathbf{r_2}$ and $\mathbf{r_3}$ for classes 2 and 3 separation is given in Equations 4.26 and 4.27. The accuracy obtained for Seeds data classification with this approach and applying 10-fold cross validation using worst-case heuristics validation split is **100%.** The parameters of the areas generated are listed in Table 5.

$$R_2 = R_{13} \,\&\, R_{23} \,\&\, ((\neg R_{14} \,\&\, \neg R_{24} \,\&\, \neg R_{33}) \,\&\, (\neg R_{34} \,\&\, \neg R_{44})) \tag{4.24}$$

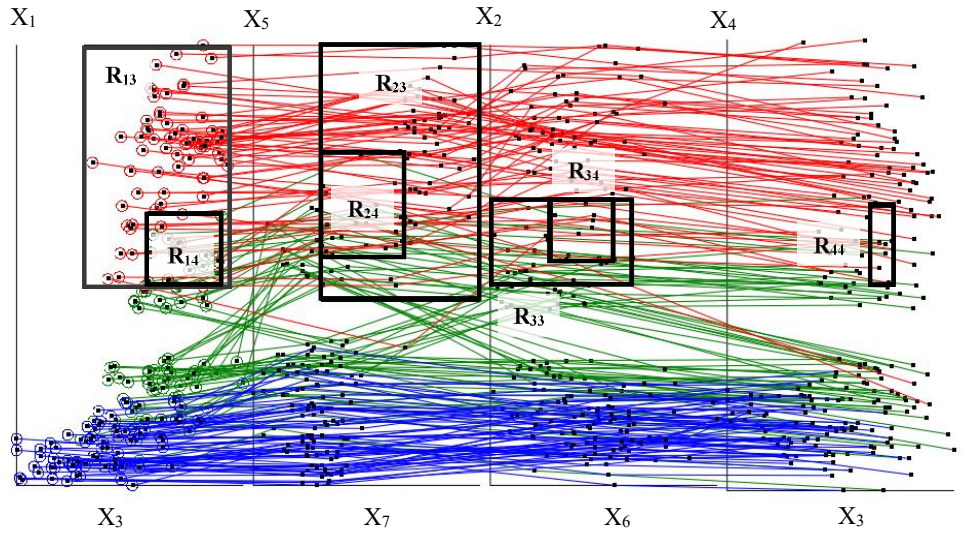
$$R_3 = (\neg R_1 \,\&\, \neg R_2) \tag{4.25}$$

$\mathbf{r_2}$: If $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) \in R_1$, then $\mathbf{x} \in$ class 2 (4.26)

$\mathbf{r_3}$: If $(x_1, x_2, x_3, x_4, x_5, x_6, x_7) \in R_3$, then $\mathbf{x} \in$ class 1 (4.27)



(a) Cases covered by rule $\mathbf{r_1}$ on Seeds dataset for class 3 (blue) separation.



(b). Cases covered by rule $\mathbf{r_2}$ on Seeds dataset for class 2 (red) separation.

FIGURE 27: Visualization of rules $\mathbf{r_1}$ and $\mathbf{r_2}$ on Seeds dataset for classes 2 and 3 separation with all the cases.

TABLE 5. Parameters of the areas generated for classification in Seeds data in 2nd iteration.

	Rectangle Parameters				
	Left	Right	Bottom	Top	Coordinate Pair
R_{13}	0.3	0.94	0.45	1.0	(X_3, X_1)
R_{14}	0.58	0.91	0.45	0.62	(X_3, X_1)
R_{23}	0.3	1.0	0.42	1.0	(X_7, X_5)
R_{24}	0.3	0.67	0.52	0.76	(X_7, X_5)
R_{33}	0.0	0.63	0.46	0.65	(X_6, X_2)
R_{34}	0.27	0.55	0.51	0.65	(X_6, X_2)
R_{44}	0.63	0.74	0.46	0.63	(X_3, X_4)

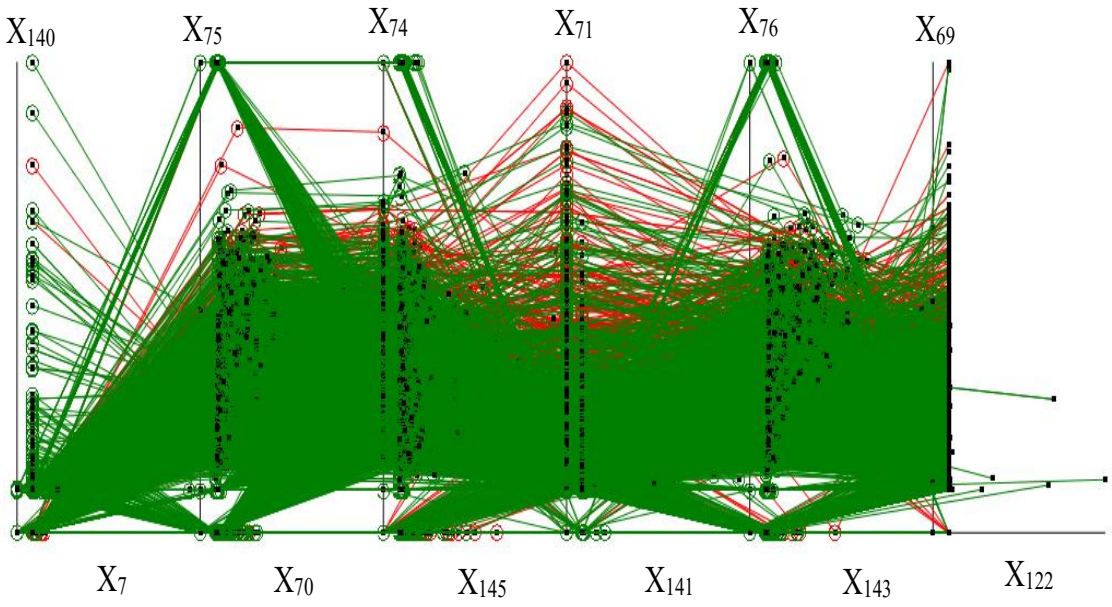
APS (Air Pressure System) Failure at Scania Trucks Data (170D)

APS failure at Scania Trucks from UCI Repository [5] consists of two classes. Class 1 corresponds to the failure in Scania trucks that is not due to the air pressure system and class 2 corresponds to the failure in Scania trucks that is due to the air pressure system. This data consists of 60000 cases and 170 dimensions. However, the dataset contains a large number of missing values. These missing values are replaced by calculating 10% of the maximum value of the corresponding column and multiplying the result by -1. In this type of imputation, all the missing data are replaced with 0 after normalization and hence does not interfere with the main data. Also, there are 4 columns of data with all 0 values. After imputation and removing the columns without any information, the final data contains 60000 cases with 166 dimensions. As discussed in Chapter II, visualizing 166 dimensions becomes very challenging. Hence, we use SCS to visualize high dimension data as shown in Figure 7.

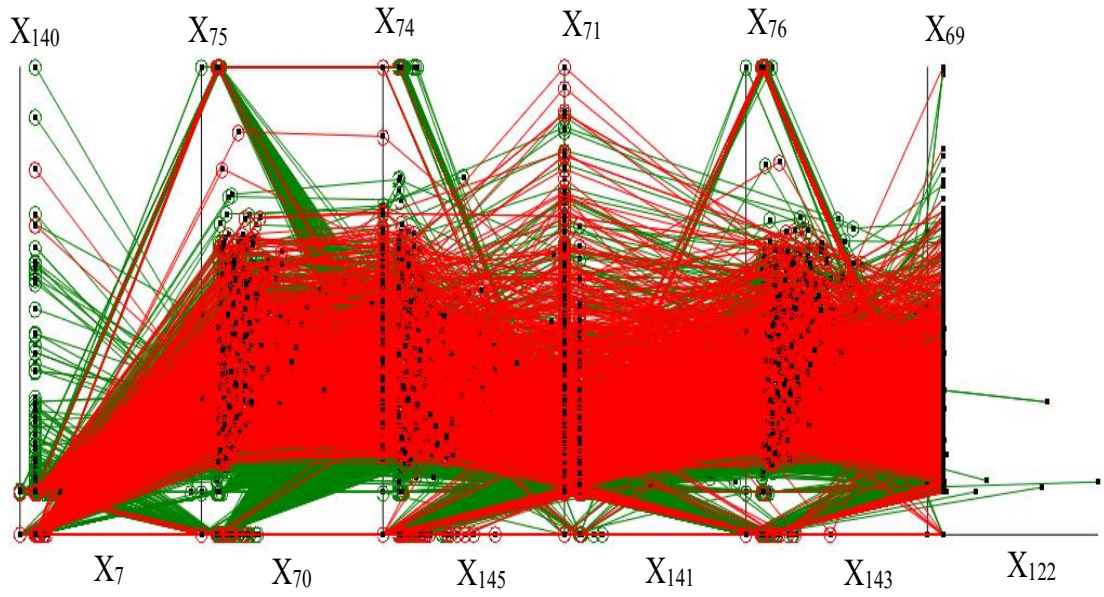
Data classification using interactive approach becomes very tedious due large

number of dimensions and instances. Hence, for this dataset, classification is performed using automation technique. After running the COO algorithm, the result contains 166 coordinates arranged from the most to least optimized coordinates. Here, we start with extracting top 4 coordinates and perform analysis on the data with 2 pairs of coordinates displayed in SPCVis. The coordinates are gradually increased by pairs until we get the desired results. In this case, 12 coordinates were selected for further analysis. They are X_7, X_{140}, X_{70}, X_{75}, X_{145}, X_{74}, X_{141}, X_{71}, X_{143}, X_{76}, X_{122}, and X_{69}. APS failure data with top 12 coordinates with green class on top is displayed in Figure 28a and red class on top is displayed in Figure 28b.

The visualizations in Figure 28 display high degree of occlusion even with the best order coordinates. Although there is fair amount of separation observed in the first pair of coordinates. Data in the remaining five pairs are highly occluded. Since there is no clear vertical separation between red class and green class, non-linear scaling becomes insignificant and hence not performed on this dataset. GA is run on this data to generate areas with high purity. The resulting visualization after running GA is displayed in Figure 29.



(a) APS failure data with green class on top.



(b) APS failure data with red class on top.

FIGURE 28: Visualization of 12 best coordinates of APS failure data in SPCVis.

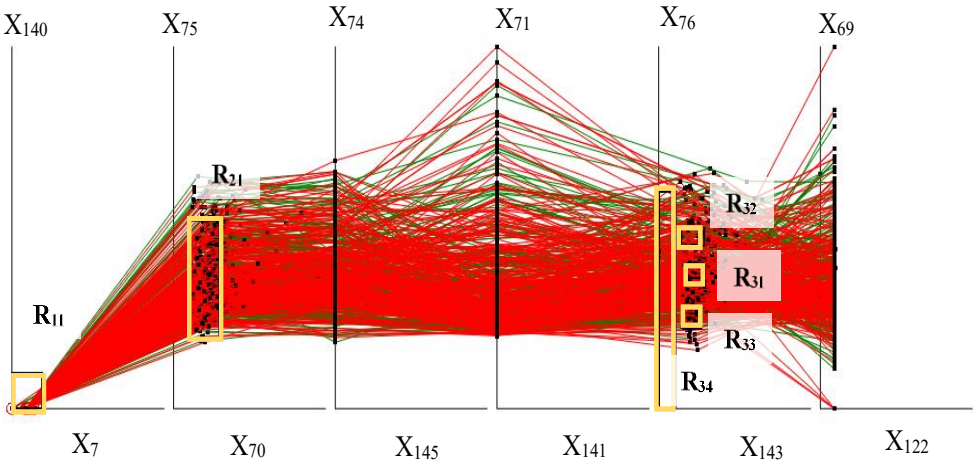
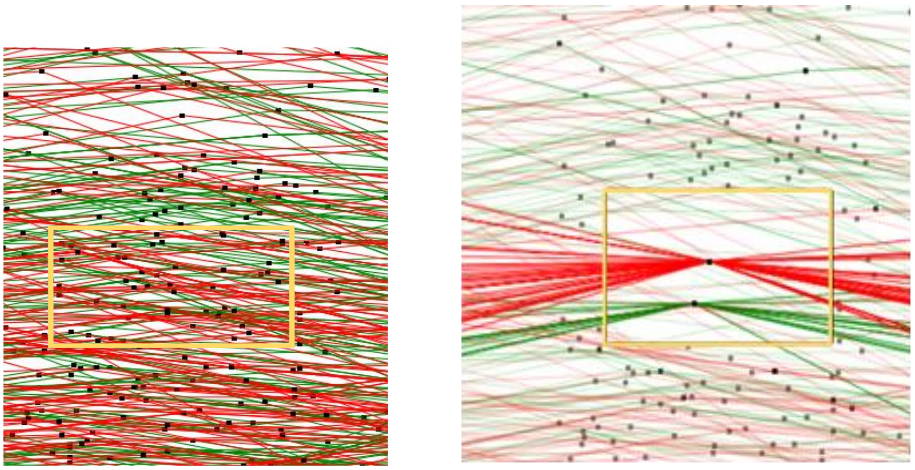


FIGURE 29: Visualization of APS failure data with areas generated by GA for red class classification.

Due to two main reasons, the data pattern cannot be interpreted by the end users in this situation: (1) high density of data within the areas generated and (2) small size of the areas generated by GA. To address these issues, we use zooming and averaging. Zoom is an interactive feature wherein the small areas can be zoomed to view the data more clearly. Figure 30 displays the zoomed image of area R_{31}.

The zoomed visualization in Figure 30a solves the problem partially. Although, the data is distinctly visible, the pattern is still hidden. To view the overall distribution of red and green class data, the average of individual class within the area is performed. Figures 30b and 31 display the averaged red and green class with R_{31} area.



(a) Visualization of zoomed R_{31} area in the APS failure data without averaging.

(b) Visualization of zoomed R_{31} area in the APS failure data after averaging.

FIGURE 30: Visualization of R_{31} area in the APS failure data with zooming and averaging.

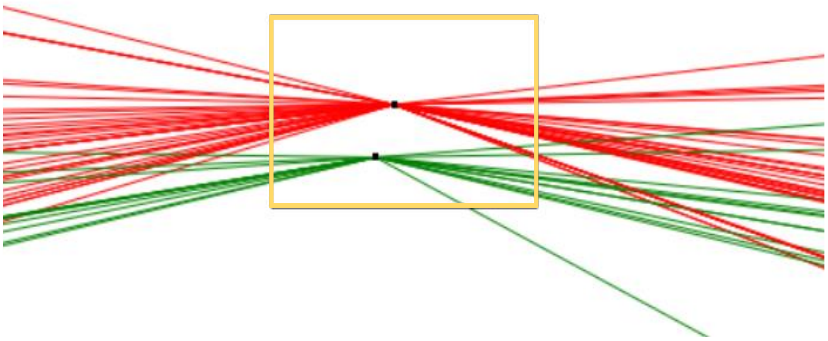


FIGURE 31: Visualization of zoomed R_{31} area in the APS failure data with averaged classes with the area (without the surrounding data).

Averaging is performed on all the areas generated by the algorithm. Since the areas are generated in the first, second and fifth pair of coordinates, we can disregard the coordinate pairs in between second and fifth, resulting in only four pairs of coordinates. The overall visualization of red class classification is displayed in Figure 32. The area R_1

rule $\mathbf{r_1}$ generated for red class 2 (red) classification is defined in Equations 4.28 and 4.29, respectively.

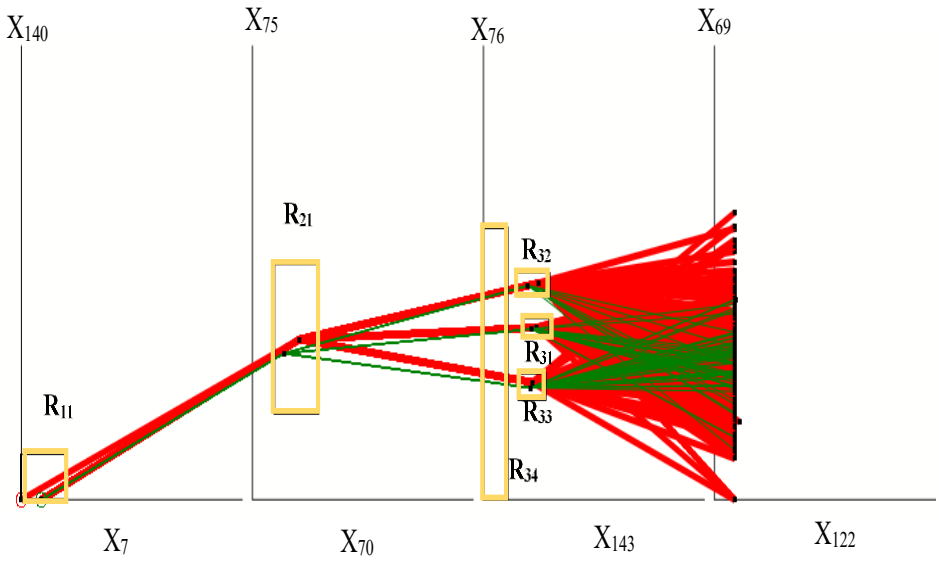
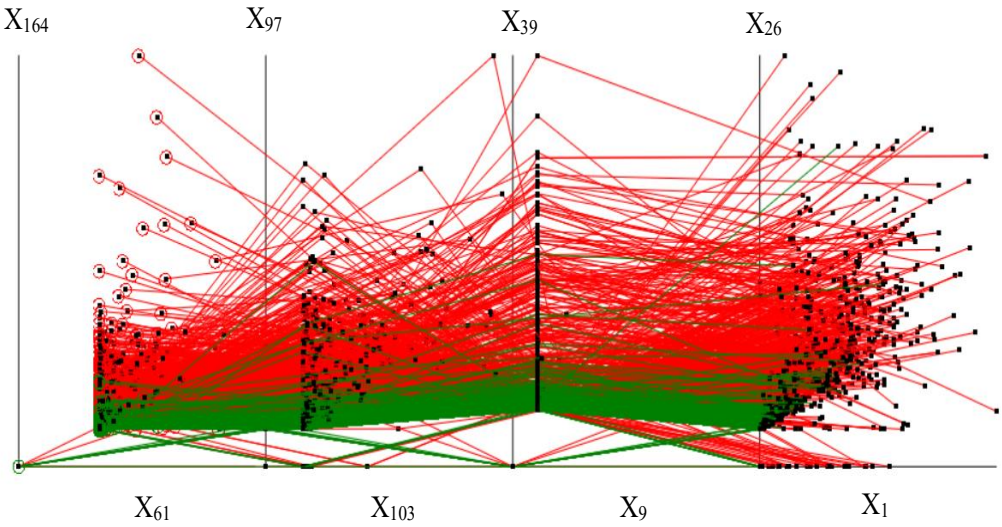


FIGURE 32: Visualization of rule $\mathbf{r_1}$ for red class classification in the APS failure data.

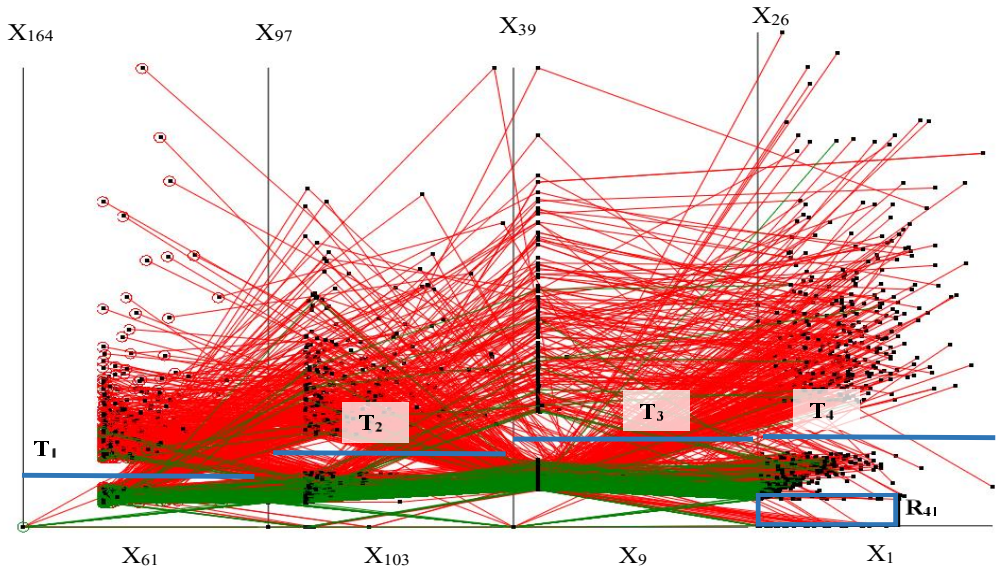
$$R_1 = R_{11} \,\&\, R_{21} \,\&\, (R_{51} \text{ or } R_{52} \text{ or } R_{53}) \,\&\, \neg(R_{54}) \tag{4.28}$$

$$\mathbf{r_1}\text{: If } (x_7, x_{140}, x_{70}, x_{75}, x_{143}, x_{76}) \in R_1 \text{ , then } \mathbf{x} \in \text{ class 2} \tag{4.29}$$

Data that do not follow R_1 rule is sent to the next iteration. The COO algorithm is run on all the 166 coordinates again to get the optimized order of coordinates for the remaining data. The order of coordinates obtained are X_{61}, X_{164}, X_{103}, X_{97}, X_9, X_{39}, X_1 and X_{26}. The data is visualized in Figure 33a.



(a) Visualization of APS failure data with top 8 coordinates.



(b) Visualization of r_2 in APS failure data with top 8 coordinates with non-linear scaling.

FIGURE 33: Visualization of APS failure data in the second iteration.

In the second iteration, the green class tends to be clustered at the bottom and red towards the top. Since the separation along vertical coordinates is clearly visible, we performed the non-linear scaling to get better data interpretation with thresholds 0.15 on

X_{164}, 0.2 on X_{97}, 0.25 on X_{39}, and 0.25 on X_{26}. Then the data is visualized with non-linear scaling and analytical rules discovered (see Figure 33b). The area R_2 and rule $\mathbf{r_2}$ for class 2 (red) classification is defined in Equations 4.30 and 4.31.

$$R_2 = T_1 \,\&\, T_2 \,\&\, T_3 \,\&\, (T_4 \text{ or } R_{41}) \tag{4.30}$$

$$\mathbf{r_2}\text{: If } (x_{61}, x_{164}, x_{108}, x_{97}, x_9, x_{39}, x_1, x_{26}) \in R_2, \text{ then } \mathbf{x} \in \text{ class 2 else class 1} \tag{4.31}$$

(T_1 & T_2 & T_3 & T_4 are the threshold values of non-linear scaling)

The accuracy obtained for APS data classification with this approach and applying 10-fold cross validation using worst-case heuristics validation split is **99.36%**. The area parameters used for APS data classification are listed in Table 6.

TABLE 6: Parameters of the areas generated for classification in APS failure data.

	Rectangle Parameters				
	Left	Right	Bottom	Top	Coordinate Pair
R_{11}	0.0	0.2	0.0	0.1	(X_7, X_{140})
R_{21}	0.1	0.3	0.19	0.52	(X_{70}, X_{75})
R_{31}	0.18	0.3	0.36	0.4	(X_{143}, X_{76})
R_{32}	0.15	0.3	0.45	0.5	(X_{143}, X_{76})
R_{33}	0.16	0.28	0.22	0.28	(X_{143}, X_{76})
R_{34}	0.0	0.11	0.0	0.6	(X_{143}, X_{76})
R_{41}	0.0	0.6	0.0	0.1	(X_1, X_{26})

CHAPTER V

EXPERIMENTAL RESULTS AND COMPARISON WITH PUBLISHED RESULTS

The results obtained are compared with the published results that uses both black-box and interpretable techniques. From Table 7, we can see that the classification accuracy obtained with the proposed method is on par with the published results and in some cases, have performed better than the published results. Since the interactive technique is more challenging for classifying data of larger size, we used only automated classification for such dataset (APS failure data). The results produced using both interactive and automated approach are listed in bold.

From the results in Table 7, we can clearly see that the accuracies obtained from our proposed method is better than black box machine learning models [21, 6, 24] and on par with interpretable models [11]. However, the accuracy for APS failure at Scania Trucks is slightly lesser compared to the accuracy in [24] using Deep Neural Network, which is a black box model. Despite of lesser accuracy, our proposed model is favorable due to its transparency, the ability to use the model as self-service and the ability to interpret the model by non-technical end users.

TABLE 7: Comparison of different classification models.

Classification Algorithms	Accuracy %
Breast Cancer data (9D)	
Iterative Visual Logical Classifier (Automated)	**99.71**
Iterative Visual Logical Classifier (Interactive)	**99.56**
SVM [3]	96.99
DCP/RPPR [16]	99.3
SVM/C4.5/kNN/Bayesian [20]	97.28
Iris Data (4D)	
Iterative Visual Logical Classifier (Automated)	**100**
Iterative Visual Logical Classifier (Interactive)	**100**
Multilayer Visual Knowledge discovery [11]	100
k-Means + J48 classifier [15]	98.67
Neural Network [21]	96.66
Seeds Data (7D)	
Iterative Visual Logical Classifier (Automated)	**100**
Iterative Visual Logical Classifier (Interactive)	**100**
Deep Neural Network [6]	100
K- nearest neighbor [19]	95.71
APS Failure at Scania Trucks (170D)	
Iterative Visual Logical Classifier (Automated)	**99.36**
Deep Neural Network (DNN) [24]	99.50
Random Forest [17]	99.02
Support Vector Machine (SVM) [17]	98.26

CHAPTER VI

CONCLUSIONS

With the help of lossless data visualization, we demonstrated the power of interpretable data classification techniques that are implemented both interactively and automatically. We observed that the interactive data classification technique works well for data with lesser cases and dimensions but fail to perform well for data with higher number of cases and dimensions. High degree of occlusion was observed and was challenging to discover pattern interactively. This issue was successfully addressed by our newly proposed automated interpretable technique using *Coordinate Order Optimizer* (COO) algorithm and *Genetic Algorithm* (GA) where the areas were generated automatically rather than interactively.

We also demonstrated the power of interactive features that improved the visualization due to which discovering patterns in the data became much easier. With non-linear scaling, zooming and averaging, the visualization was improved to increase data interpretability. The SPCVis software successfully visualized the larger dataset using SCS. Our proposed techniques can be further leveraged by incorporating more interactive features like non-orthogonal coordinates, data reversing etc. SPC and SCS visualizations help us to discover only specific patterns in the data and our future goal is to incorporate more General line coordinate visualizations and also to provide a medium for visualizing thousands of dimensions with millions of data cases.

REFERENCES CITED

[1] Banzhaf W, Nordin P, Keller RE, Francone FD. Genetic programming: an introduction. San Francisco: Morgan Kaufmann Publishers; 1998 Jan.

[2] Bouali F, Serres B, Guinot C, Venturini G. Optimizing a radial visualization with a genetic algorithm. In2020 24th International Conference Information Visualisation (IV), 2020 Sep 7 , pp. 409-414. IEEE.

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

[4] Cowgill MC, Harvey RJ, Watson LT. A genetic algorithm approach to cluster analysis. Computers & Mathematics with Applications. 1999 Apr 1;37(7):99-108.

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

[6] Eldem A. An application of deep neural network for classification of wheat seeds. European Journal of Science and Technology, No 19 (pp. 213-220), August 2020.

[7] Everitt B. The Cambridge Dictionary of Statistics. Cambridge University press. Cambridge, UK Google Scholar, 1998.

[8] 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.

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

[10] Kovalerchuk, B., Ahmad, M.A., Teredesai A., Survey of explainable machine learning with visual and granular methods beyond quasi-explanations, In: Interpretable Artificial Intelligence: A Perspective of Granular Computing, W. Pedrycz, S.M.Chen ((Eds.), pp. 217-267, 2021.

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

[12] Kovalerchuk B., Gharawi A., Decreasing occlusion and increasing explanation in interactive visual knowledge discovery, In: Human Interface and the Management of

Information. Interaction, Visualization, and Analytics, Lecture Notes in Computer Science series, Vol. 10904, 2018, pp. 505-526. Springer.

[13] Kovalerchuk B. Enhancement of cross validation using hybrid visual and analytical means with Shannon function. In: Beyond Traditional Probabilistic Data Processing Techniques: Interval, Fuzzy etc. Methods and Their Applications, 2020 (pp. 517-543). Springer.

[14] Kovalerchuk, B., Grishin, V. Reversible data visualization to support machine learning, In: Human Interface and the Management of Information. Interaction, Visualization, and Analytics, Lecture Notes in Computer Science series, Vol. 10904, 2018, pp. 45-59, Springer.

[15] 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.

[16] Neuhaus, N., Kovalerchuk, B., Interpretable machine learning with boosting by Boolean algorithm, Joint 2019 8th Intern. Conf. on Informatics, Electronics & Vision (ICIEV) & 3rd Intern. Conf. on Imaging, Vision & Pattern Recognition (IVPR), Spokane, WA, 2019, 307-311.

[17] Rafsunjani S, Safa RS, Al Imran A, Rahim MS, Nandi D. An empirical comparison of missing value imputation techniques on APS failure prediction. IJ Inf. Technol. Comput. Sci. 2019 Feb;2:21-9.

[18] Rifki O, Ono H. A survey of computational approaches to portfolio optimization by genetic algorithms. In18th International Conference Computing in Economics and Finance 2012 Jun. Society for Computational Economics.

[19] Sabanc 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.

[20] Salama GI, Abdelhalim M, Zeid MA. Breast cancer diagnosis on three different datasets using multi-classifiers. International Journal of Computer and Information Technology, Vol. 01– Issue 01, 2012.

[21] 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.

[22] Wagle, S**,** SPCVis, Interactive Shifted Paired Coordinate Visualization Tool. `https://github.com/Wagle1/SPCVis.git`, 2021. Accessed: 2021-05-30

[23] Wagle, S., Kovalerchuk, B., Interactive visual Self-Service data classification approach to democratize machine learning , 24th International Conference Information Visualisation IV-2020, Melbourne, Victoria, Australia, 7-11 Sept.2020, IEEE, DOI 10.1109/IV51561.2020.00052.

[24] Zhou F, Yang S, Fujita H, Chen D, Wen C. Deep learning fault diagnosis method based on global optimization GAN for unbalanced data. Knowledge-Based Systems. 2020 Jan 1;187:104837.

APPENDIX

SPCVIS MANUAL

SPCVis provides a medium for visualizing multidimensional (n-D) data in SPC without losing any information. This system is developed on Windows Forms in C++ and OpenGL library. The user can load the data using 'Upload Data' option and visualize the data in SPC without any loss of information. Apart from the lossless representation of the n-D data, the SPCVis also provides user interactive controls like clicking and dragging the user selected graphs on the screen and reversing the user selected coordinates of the data to reorient the data representation according to user convenience. These features help in better understanding of the data, especially when the data contain two or more classes. Additional options like zooming and panning are provided for user convenience for data exploration. Also, color selection functionality is given where the user can customize the color of data classes. The SPCVis software can be downloaded from the GitHub repository mentioned in [22]. The screenshot of the SPCVis software with WBC data is shown in Figure 34.

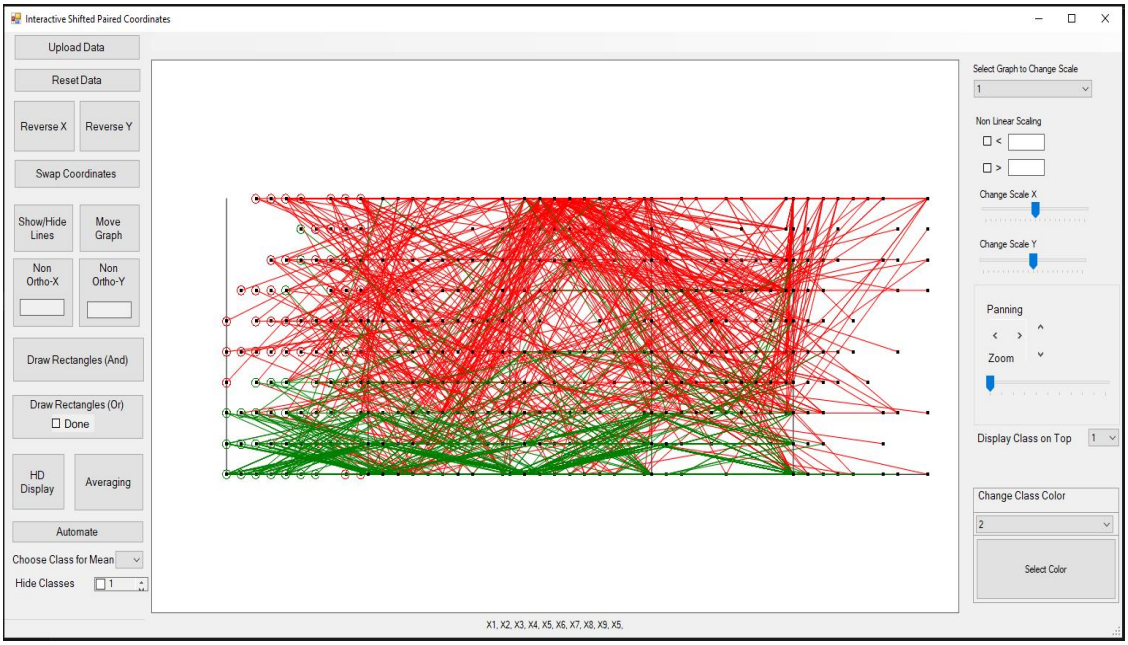
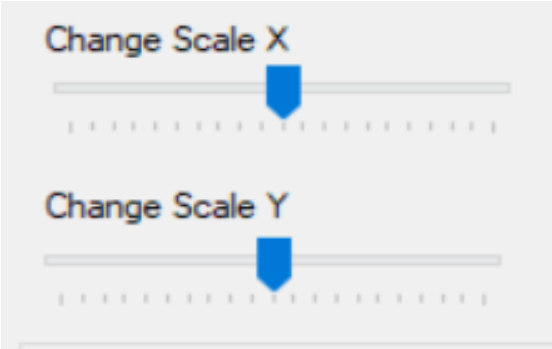
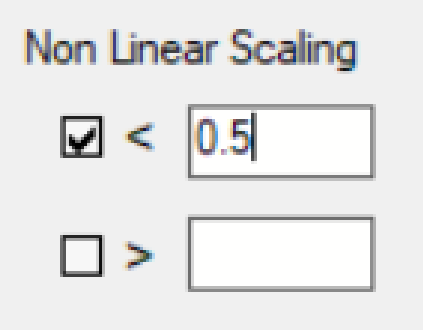


FIGURE 34: SPCVis software displaying WBC data.

Figure 35a displays the implementation of non-linear scaling feature. It is implemented using separate sliders for x axis and y axis. Figure 35b displays the user interface to enter the *k* value for non-linear scaling (see Equation 2.1). The output of non-linear scaling is displayed in Figure 4. Non-orthogonal coordinate system has a coordinate inclined at an angle other than 90 degrees with respect to the other coordinate. User interactive implementation of non-orthogonal coordinate is shown in Figure 36. This interactive feature is implemented using a button and a text entry box where the user can enter the angle at which the inclination is to be displayed. The outputs of non-orthogonal coordinates are displayed in Figures 5 and 6.



(a) Sliders for performing non-linear scaling.



(b) User Interface for entering *k* value in non-linear Scaling.

FIGURE 35: Implementation of non-linear scaling in SPCVis.

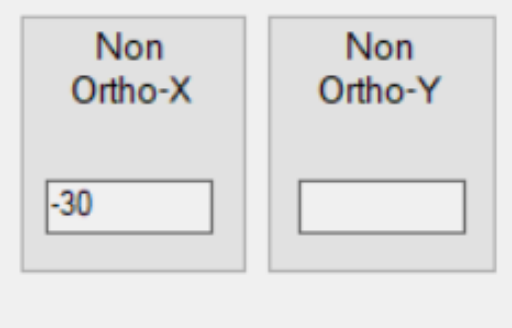


FIGURE 36: User Interface for Non-Orthogonal Coordinates.

The rectangular areas drawn using the option ‘Draw Rectangle’ provided in the SPCVis software. This allows the user to draw rectangles on any coordinate pair using mouse (In order to draw the rectangles that follow OR logical rule, ensure to check the checkbox below ‘Draw Rectangle (Or)’ button after all the rectangles are drawn). See Figures 9 -12 for the outputs with rectangles drawn interactively. Use the ‘HD Display’ (High Dimension Display) to view the data with higher dimensions. This option can be used if data contains more than 30 dimensions (see Figure 7).