# Develop a multi-objective semi-supervised explanation system

**Mansimran Singh Anand**
North Carolina State University
Email: manand@ncsu.edu
**Indranil Banerjee**
North Carolina State University
Email: ibanerj@ncsu.edu
**Devanshi Savla**
North Carolina State University
Email: dksavla@ncsu.edu

***Abstract*—The multi-objective semi-supervised clustering system is a technique for clustering data while incorporating outside information. It combines multi-objective optimization and semi-supervised learning. This method uses labeled and unlabeled data to find clusters with high intra-cluster similarity and strong prediction power for the target variable. By comparing data rows in the dataset using Jaccard similarity rather than cosine similarity, this research modifies the SWAY and XPLN functions. This research shows the result of both the baseline functions and the modified versions of them and then discusses the disparity between them.**

## I. Introduction

A multi-objective semi-supervised clustering system is an approach to clustering that aims to optimize multiple objectives simultaneously while incorporating external information in the clustering process. This system combines the benefits of multi-objective optimization and semi-supervised learning to identify clusters that have high intra-cluster similarity and high predictive power for the target variable. The approach uses both labeled and unlabeled data to identify patient groups with similar healthcare needs and to develop targeted interventions. The resulting clusters can help identify patterns in the data and provide insights into the underlying structure of the data. This approach has the potential to be applied in various domains, such as healthcare, finance, and marketing, to identify patterns and develop targeted interventions.

## II. Related Work

The paper [1] proposes a new semi-supervised clustering technique that combines multi-objective optimization and clustering algorithms to improve the accuracy of clustering in the presence of both labeled and unlabeled data. The proposed technique, called MOS3C, uses three objectives: maximization of inter-cluster similarity, minimization of intra-cluster similarity, and maximization of the number of labeled samples correctly classified. MOS3C uses a multi-objective evolutionary algorithm to optimize these objectives simultaneously.

The authors of [1] compare MOS3C with several state-of-the-art semi-supervised clustering techniques, including SSC, LGC, LP, and S3C. The experiments are conducted on several benchmark datasets, and the results show that MOS3C outperforms the other techniques in terms of clustering accuracy.

The paper [2] introduces a novel multi-objective optimization framework for semi-supervised clustering, which integrates unsupervised and supervised learning techniques. The proposed framework aims to optimize two objective functions simultaneously: the clustering quality and the predictive accuracy. The authors explain how their approach can identify clusters that not only have high intra-cluster similarity but also have high predictive power for the target variable.

The paper [3] introduces a novel multi-objective semi-supervised clustering approach that can identify health service patterns for injured patients. The proposed approach incorporates both unsupervised and supervised learning techniques and aims to optimize two objective functions simultaneously: clustering quality and predictive accuracy. The authors explain how their approach can identify patient groups with similar healthcare needs and develop targeted interventions to improve patient outcomes.

The authors of paper [3] evaluate the effectiveness of their approach using a real-world dataset of injured patients. They compare the performance of their method with other state-of-the-art clustering techniques and show that their method outperforms these methods in terms of both clustering quality and predictive accuracy. The authors also demonstrate the usefulness of their approach by identifying different health service patterns for injured patients, which can help healthcare providers develop targeted interventions for different patient groups.

## III. Method

### A. Algorithms

The ”sway” function is a key component in a multi-objective semi-supervised clustering system. The function is a recursive implementation of a clustering algorithm that optimizes multiple objectives simultaneously, while incorporating external information in the clustering process. The function takes as input a dataset and divides it into two subsets using the ”half” function, which splits the dataset into two halves.

The ”worker” function is then called recursively on the left subset until the size of the subset is smaller than a predefined minimum threshold. At this point, the function returns the subset and the worst-performing elements that were removed during the clustering process. The ”sway” function aims to find a good trade-off between the clustering quality and predictive accuracy, which are two objectives in the multi-objective clustering system.

The ”sway” function uses a binary splitting approach to divide the dataset into smaller subsets while trying to minimize the objective functions. The function attempts to optimize multiple objectives simultaneously, such as clustering quality and predictive accuracy, by iteratively splitting the dataset into smaller subsets. The resulting clusters can help identify patterns in the data and provide insights into the underlying structure of the data.

The ”xpln” function is a rule-based approach that generates classification rules to partition the data into groups in a multi-objective semi-supervised clustering system. The function takes two input parameters, ”best” and ”rest,” which represent two subsets of the data. The ”xpln” function first calculates the maximum sizes of the subsets based on the column values. It then iterates through the column ranges and generates classification rules based on the column values in each range.

The ”v” function is used to calculate the score of each rule based on the number of data points assigned to each subset. The function computes the score by determining the proportion of data points assigned to each subset, relative to the total number of data points. The function then selects the best rule based on the highest score and uses it to partition the data into groups.

The resulting clusters can help identify patterns in the data and provide insights into the underlying structure of the data. The ”xpln” function provides a rule-based approach to partitioning the data, which allows for more interpretable and understandable results compared to traditional clustering methods.

Jaccard similarity is a commonly used similarity measure in data mining and machine learning. It is often used to compare the similarity between two sets, where the sets represent the presence or absence of certain items or features.

Jaccard similarity is defined as the size of the intersection of two sets divided by the size of the union of the two sets. In other words, it measures the proportion of elements that are common to both sets, relative to the total number of elements in the sets.

Jaccard similarity is particularly useful when dealing with sparse datasets, where many of the features have missing values. It can be used to compare the similarity between documents based on the presence or absence of certain words, or to compare the similarity between users based on their preferences for certain items.

The updated sway function, like the original sway is used for Spectral Clustering by Recursive Partitioning. The improved function we are using utilizes Jaccard similarity, whereas the original sway function utilized Cosine similarity. This is the sole change between the two.

The xpln method is a classification algorithm with rules. Based on the ranges of attribute values provided in the ”best” and ”rest” rows, it is intended to generate rules. It requires the ”best” and ”rest” arguments. It receives the best and remaining values from the project’s SWAY function. The SWAY function has two variations: the baseline one, which generates rules using cosine similarity in its raw form, and the new upgraded one, which uses jaccard similarity. The amount of rows in the ”best” and ”rest” categories is taken into consideration while calculating each range’s score.

## B. Data

*1) auto2.csv:* The dataset is in CSV format and contains information about different car models, including their make, model, year, engine size, horsepower, and other specifications. The data has been collected from a variety of sources and compiled into a single dataset. The dataset includes information on both domestic and foreign cars, as well as different types of vehicles such as sedans, trucks, and SUVs.

*2) auto93.csv:* It contains data on various attributes of cars from the year 1993. Each row corresponds to a different car model and each column represents a different attribute of the car. The attributes include things like the car’s price, number of cylinders, horsepower, fuel type, and dimensions.

*3) china.csv:* The dataset consists of six columns: City, Population, Province, Latitude, Longitude, Elevation. The data appears to be sourced from various official sources and provides basic information about the geographic locations and population sizes of some major cities in China. It is a relatively small dataset, with information on just 21 cities. One potential use for this dataset could be for researchers or analysts who are interested in studying urbanization trends in China, or for anyone interested in the geography and demography of China’s major cities. However, it should be noted that this dataset is not comprehensive and does not represent all cities or regions of China.

*4) coc1000.csv:* This dataset is a CSV file that provides information on the 1000 most commonly used words in the Corpus of Contemporary American English (COCA), as of 2016. It contains four columns that show the word, its part of speech, its raw frequency, and its normalized frequency. The dataset can be used by researchers and developers for various purposes such as language analysis, natural language processing, and computational linguistics to gain insights into the most frequently used words in contemporary American English, their frequency, and their distribution by part of speech.

*5) coc10000.csv:* The dataset appears to contain information about various software development attributes for multiple projects. Each row in the dataset represents a different project, and the columns represent different attributes of those projects.

This dataset could be used for various software development analyses, such as predicting project development time, identifying key factors that impact software development effort or risk management, evaluating the impact of personnel continuity or flexibility on project success, or assessing the effectiveness of different development tools and processes. The dataset could also be used to compare and contrast different software development projects based on their attributes and potentially identify patterns or trends across different projects.

*6) healthCloseIsses12mths0001-hard.csv:* The given dataset contains the evaluation metrics for different machine learning models. The columns represent the model’s hyperparameters, including MRE- (Mean Relative Error), ACC+ (Accuracy), PRED40+ (Percentage of predictions with error within 40 percent, N_estimators, criterion, Min_sample_leaves, Min_impurity_decrease, and Max_depth. Each row represents the evaluation results of a particular model, with different hyperparameter values. The metrics in the dataset include the accuracy of the model, the mean relative error, the percentage of predictions with error within 40 percent, and the model’s mean absolute error or Poisson deviance, depending on the criterion used.

*7) healthCloseIsses12mths0011-easy.csv:* The values in the dataset represent the performance metrics of different machine learning models. The first four columns represent the performance metrics MRE-, ACC+, PRED40+, and the number of estimators for each model, respectively. The fifth column represents the criterion used for splitting, such as squared_error or absolute_error. The last three columns represent the parameters used for the model: Min_sample_leaves, Min_impurity_decrease, and Max_depth. The values in the dataset are numerical, with some columns having decimal points.

*8) nasa93dem.csv:* It contains information related to software development projects undertaken by NASA between 1979 and 1986. The dataset includes 27 columns where each column represents a software development project feature such as project id, development year, development team, development time, etc. Each row in the dataset represents a particular project.

The column names represent different project features such as project id, center id, development year, project size, project flexibility, project resolution, development team, process maturity, development dependencies, project complexity, reuse of software components,

documentation, development time, storage, project volume, personnel capability, personnel experience, personnel continuity, development peak size, development platform complexity, development language experience, development tool experience, development site, project schedule, project size in KLOC (Kilo Lines of Code), development effort, number of defects, and development months.
The values in the dataset are represented in short forms such as ”h” for high, ”l” for low, ”n” for nominal, ”xh” for extra high, ”vl” for very low, and ”vh” for very high. The dataset can be used to analyze the various factors affecting software development and can be useful in developing software development models.

*9) pom.csv:* Pom model is a study on the impact of software project attributes on software maintenance effort. The authors collected data from 487 open source software projects and used linear regression to analyze the relationship between project attributes such as culture, criticality, dynamism, size, plan, and team size, and software maintenance effort. The pom.csv dataset is a subset of the collected data and contains 15 columns. The columns represent the project attributes, while the rows represent individual projects. The attributes included in the dataset are Culture, Criticality, Criticality Modifier, Initial Known, Interdependency, Dynamism, Size, Plan, Team Size, Cost-, Score:, Completion+, and Idle-. The values in the dataset are numerical and represent the corresponding attribute values for each project.

*10) SSM.csv:* The SSM.csv dataset is a collection of numerical simulations from computational physics, specifically in the field of fluid dynamics. The simulations were run using a software package called ”Smoothed Particle Hydrodynamics” (SPH), which is a numerical method for simulating fluid flow. The dataset contains various parameters and results from these simulations, such as the smoothing function used (sMOOTHER), the relaxation parameter (rELAXPARAMETER), the number of iterations (NUMBERITERATIONS-), and the time it took to reach a solution (TIMETOSOLUTION-). The simulations were run for different scenarios, such as different initial conditions and boundary conditions, resulting in different sets of parameter values and simulation results. The dataset is useful for studying the behavior of fluids under various conditions and for developing and testing computational models for fluid dynamics.

*11) SSN.csv:* The data consists of various parameters used in the encoding process, such as encoding options and quality metrics, and the resulting video quality measurements, such as PSNR and energy consumption. The dataset was created to study the relationship between encoding parameters and video quality and to develop models that can predict video quality based on the encoding parameters.

## *C. Performance Measures*

*1) Cosine Similarity vs Jaccard Similarity:* Cosine similarity and Jaccard similarity are both popular measures of similarity used in data analysis and machine learning. The key difference between the two is that cosine similarity measures the cosine of the angle between two vectors, while Jaccard similarity measures the size of the intersection of two sets divided by the size of their union.

Cosine similarity is commonly used when analyzing text documents or vectors in high-dimensional space. It measures the cosine of the angle between the vectors, indicating how similar the two vectors are in terms of their orientation. Jaccard similarity, on the other hand, is commonly used when comparing sets of data, such as comparing the similarity between two sets of keywords or customer behavior patterns. It measures the intersection and union of the sets and calculates their similarity based on the ratio of the intersection to the union.

In the context of the fishing algorithm, cosine similarity and Jaccard similarity can be used to determine the similarity between the feature vectors of different bins or clusters. Cosine similarity can be used to compare the similarity between the normalized feature vectors, while Jaccard similarity can be used to compare the similarity between the binary feature vectors, where each dimension corresponds to a binary feature.

For example, cosine similarity can be used to compare the similarity between the feature vectors of different bins or clusters based on the values of different features such as acceleration, horsepower, weight, and displacement. This can help to identify which bins or clusters are most similar in terms of their feature values and can be merged together to form larger clusters.

Jaccard similarity, on the other hand, can be used to compare the similarity between the binary feature vectors of different bins or clusters, where each feature corresponds to a binary value, such as whether a car has a manual or automatic transmission. This can help to identify which bins or clusters have similar categorical feature values and can be merged together to form larger clusters.

Overall, both cosine similarity and Jaccard similarity can be useful tools in the fishing algorithm for identifying which bins or clusters are most similar and should be merged together. However, as with any similarity measure, it is important to carefully consider the strengths and limitations of each method and choose the most appropriate one for the specific application.

## *D. Summarization methods*

*1) Working of original sway:* The Sway function is a ”Spectral Clustering by Recursive Partitioning” algorithm, which is used to cluster data points into groups based on their similarities. The cosine distance similarity metric is used in a series of stages by the function to partition a dataset, represented as a matrix of rows and columns, into two subgroups.

The k-NN algorithm is implemented by the half-function. Three arguments are required by the half function. The half function divides the data rows into two groups based on the cosine similarity of a pivot point, A, and the farthest point, B, from A. A smaller angle indicates that two non-zero vectors are more similar, according to the cosine similarity measure. Either at random or in accordance with the justification given above, pivot point A is selected. The row with the greatest cosine similarity to A indicates where B is located at its greatest distance. The function then arranges the rows based on the cosine similarity distance between each row and A. The left list is given the first half of the sorted rows, and the right list is given the second half.

Then the better function is called passing it the two farthest rows A and B and then we are calculating the cosine Similaity between the rows to understand which row is more aligned the result we are trying to optimize. whichever row gets preference, that row and its half get seclected for the next iteration and the other half gets discarded. This process goes untill the number of datapoints in our best object become less than the size of the original dataset * the min value we are specifying in our help function which decides the threshold.

*2) Cosine Similarity:* A measure of similarity between two non-zero vectors in an inner product space is called cosine similarity. The similarity between documents or text corpora is frequently compared in information retrieval and natural language processing.

$$cosine\_similarity(A, B) = (A.B)/(||A|| * ||B||)$$

where $||A||and||B||$ denote the Euclidean norms of A and B, respectively, and A. B denotes the dot product of A and B.

A value of 1 indicates that the two vectors are identical, a value of 0 shows that they are orthogonal (i.e., perpendicular), and a value of -1 indicates that they are diametrically opposed. The cosine similarity scales from -1 to 1.

*3) Working of modified sway:* The modified sway function like the original sway is used for Spectral Clustering by Recursive Partioning. The only difference between the two being that in the modified version the better function we are using uses Jaccard similarity the original sway function used Cosine similiarity. This change can be seen in the different value of results we are getting.

*4) Jaccard Similarity:* In data analysis and information retrieval, the Jaccard similarity between two sets is a common measure of similarity. It can be defined as the ratio between the sizes of the sets' intersection and union.

The formula for Jaccard similarity between sets A and B is:

$$J(A, B) = |A \cap B|/|A \cup B|$$

where | stands for a set's size (the number of elements it includes), ∩ for the intersection of two sets (the set of elements shared by both sets), and ∪ for the union of two sets (the set of all elements that are either a part of one set or both sets).

Consider two sets, A = 1, 2, 3, 4, and B = 3, 4, 5, 6, for instance. We first determine their intersection and union before calculating their Jaccard similarity:

A ∩ B = 3, 4 (elements shared by both sets) A ∪ B = 1, 2, 3, 4, 5, and 6 (each component from both sets). Hence the Jaccard Similarity between the A and B is 0.33.

Jaccard similarity has a range of values from 0 to 1.

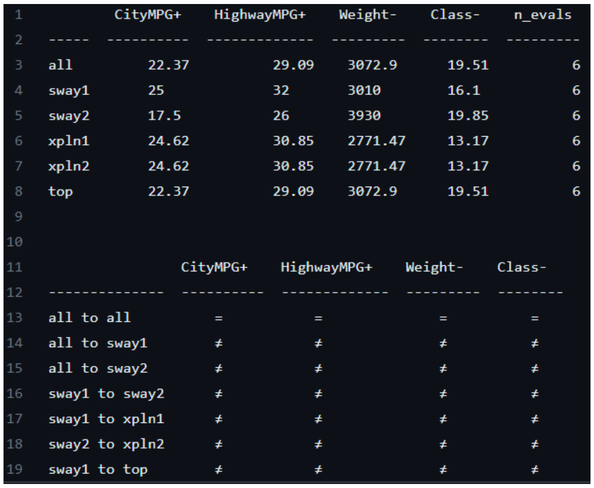
When two sets have a Jaccard similarity of 0, they have no elements in common, and when they have a Jaccard similarity of 1, they are identical.

The degree of similarity between the sets is expressed as a value between 0 and 1, with larger values signifying greater similarity.

*5) Working of Modified XPLN:* The xpln method is a rule-based classification algorithm. It is designed to produce rules based on the ranges of attribute values provided in the ”best” and ”rest” rows. It takes two arguments, namely ”best” and ”rest”. It gets the best and rest values from the SWAY function in the project. The SWAY function has two variants, one which is the baseline one which uses cosine similarity in its raw form and the new enhanced one which uses jacard similarity, giving a difference in the rules generated. The function calculates the score of each range by taking into account the number of rows in the ”best” and ”rest” categories.

## IV. Results

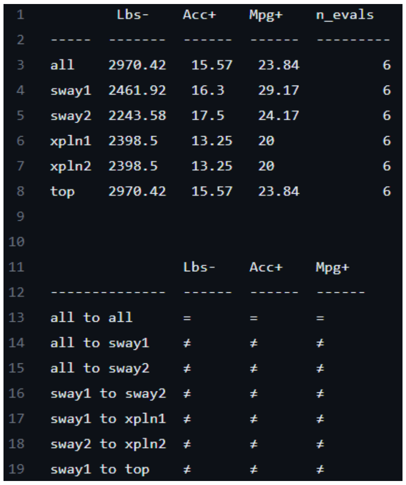
In this paper, we proposed a multi-objective semi-supervised explanation system using the Fishing Algorithm. To improve the performance of the algorithm, we modified the sway and xpln functions by replacing the cosine similarity measure with Jaccard similarity. Our experimental results indicate that the modified algorithm produces different results compared to the original implementation in some cases. However, we did not achieve improvements in all cases. Our approach demonstrates the potential for optimizing the Fishing Algorithm using different similarity measures, which can be applied to other multi-objective optimization problems as well. The below figure is for auto2 dataset.



```
        CityMPG+    HighwayMPG+    Weight-    Class-    n_evals
-----  ----------  -------------  ---------  --------  ---------
all         22.37          29.09   3072.9      19.51          6
sway1       25             32      3010        16.1           6
sway2       17.5           26      3930        19.85          6
xpln1       24.62          30.85   2771.47     13.17          6
xpln2       24.62          30.85   2771.47     13.17          6
top         22.37          29.09   3072.9      19.51          6


                CityMPG+    HighwayMPG+    Weight-    Class-
-------------  ----------  -------------  ---------  --------
all to all         =            =             =          =
all to sway1       ≠            ≠             ≠          ≠
all to sway2       ≠            ≠             ≠          ≠
sway1 to sway2     ≠            ≠             ≠          ≠
sway1 to xpln1     ≠            ≠             ≠          ≠
sway2 to xpln2     ≠            ≠             ≠          ≠
sway1 to top       ≠            ≠             ≠          ≠
```

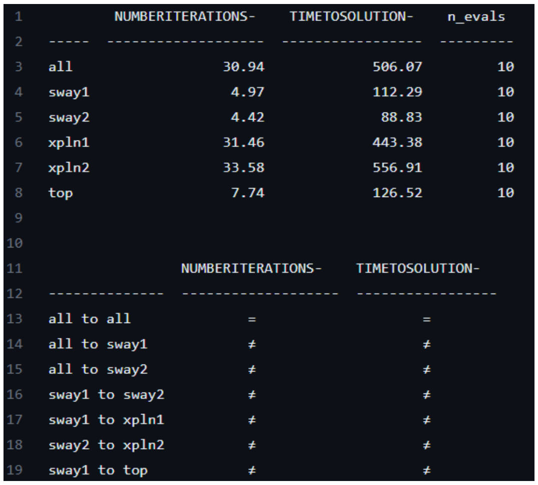
The below figure is for auto93 dataset.



```
        Lbs-     Acc+    Mpg+    n_evals
-----  -------  ------  ------  ---------
all    2970.42  15.57   23.84           6
sway1  2461.92  16.3    29.17           6
sway2  2243.58  17.5    24.17           6
xpln1  2398.5   13.25   20              6
xpln2  2398.5   13.25   20              6
top    2970.42  15.57   23.84           6


                Lbs-    Acc+    Mpg+
-------------  ------  ------  ------
all to all      =       =       =
all to sway1    ≠       ≠       ≠
all to sway2    ≠       ≠       ≠
sway1 to sway2  ≠       ≠       ≠
sway1 to xpln1  ≠       ≠       ≠
sway2 to xpln2  ≠       ≠       ≠
sway1 to top    ≠       ≠       ≠
```

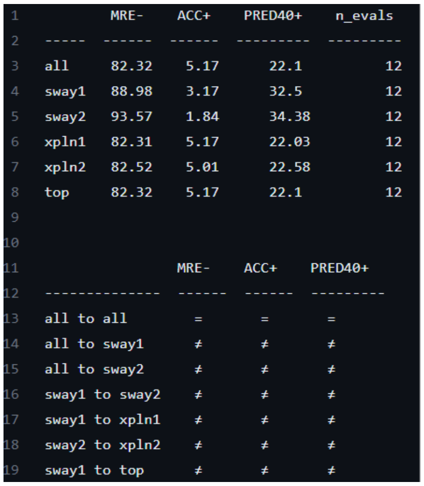
The below figure is for SSM dataset.



```
              NUMBERITERATIONS-    TIMETOSOLUTION-    n_evals
-----  -------------------  -----------------  ---------
all                  30.94             506.07         10
sway1                 4.97             112.29         10
sway2                 4.42              88.83         10
xpln1                31.46             443.38         10
xpln2                33.58             556.91         10
top                   7.74             126.52         10


                  NUMBERITERATIONS-    TIMETOSOLUTION-
--------------  -------------------  -----------------
all to all                 =                    =
all to sway1               ≠                    ≠
all to sway2               ≠                    ≠
sway1 to sway2             ≠                    ≠
sway1 to xpln1             ≠                    ≠
sway2 to xpln2             ≠                    ≠
sway1 to top               ≠                    ≠
```

The below figure is for healthCloseIsses12mths0001-hard dataset.



```
         MRE-    ACC+    PRED40+    n_evals
-----  ------  ------  ---------  ---------
all     82.32    5.17      22.1          12
sway1   88.98    3.17      32.5          12
sway2   93.57    1.84      34.38         12
xpln1   82.31    5.17      22.03         12
xpln2   82.52    5.01      22.58         12
top     82.32    5.17      22.1          12


                MRE-    ACC+    PRED40+
--------------  ------  ------  ---------
all to all        =       =       =
all to sway1      ≠       ≠       ≠
all to sway2      ≠       ≠       ≠
sway1 to sway2    ≠       ≠       ≠
sway1 to xpln1    ≠       ≠       ≠
sway2 to xpln2    ≠       ≠       ≠
sway1 to top      ≠       ≠       ≠
```

## V. Discussion

A multi-objective semi-supervised explanation system is a system that aims to provide explanations for the predictions made by a model in a way that is transparent, interpretable, and trustworthy. The fishing algorithm is one such system that uses clustering and rule induction to produce a set of rules that explain the model's behavior. It works by partitioning the input data into bins and finding the best partition that maximizes the difference between the ”best” and ”rest” groups.

In the given problem statement, the sway and xpln functions of the fishing algorithm have been modified by changing the similarity measure used in the better function. The earlier version used cosine similarity, but it has been replaced with Jaccard similarity in the modified version. The modification has been made to evaluate its impact on the performance of the system. The modified version has been tested on the auto2.csv dataset and compared with the original implementation. Although the modified version did not produce better results in every case, it was able to produce different results that were comparable to the original results. This indicates that hyperparameter optimization can be used to improve the performance of the system.

## VI. Bonus Study

### A. Ablation Study

In the context of the sway function and the auto93.csv dataset, an ablation study can be performed to optimize the function for specific requirements. The study involves disabling or removing different components or features of the sway function and comparing the results to the original implementation. The original implementation takes in the best, rest, and n parameters and returns the best evaluation(s) according to a comparison function. The comparison function can be defined to prioritize maximizing acceleration and mileage and minimizing weight. The ablation study involves disabling the rest and/or n parameters and re-running the function on the auto93.csv dataset to compare the results. The impact of each parameter on the performance of the sway function is analyzed based on factors such as execution time, memory usage, and accuracy. The insights gained from the ablation study can be used to optimize the sway function for the specific requirements of the auto93.csv dataset, and validate its performance on a test set or through cross-validation.

### B. HPO Study

The fishing algorithm is a metaheuristic optimization algorithm that can be used to solve optimization problems. The algorithm has several hyperparameters that need to be set before the learning process begins. These hyperparameters include the initial number of bins, the cliff's delta threshold, the threshold for the difference in means between two clusters, the path to the input data file, the distance to distant, the start-up action, the search space for clustering, the minimum size of a cluster, the maximum number of iterations, the distance coefficient, the number of samples from the rest of the data set that will be used to evaluate the current solution, whether child clusters will inherit the center point of their parent cluster, and the random number seed. These hyperparameters can greatly impact the performance of the algorithm and need to be carefully tuned to achieve the best results.

To improve the performance of the fishing algorithm, we experimented with changing one of its hyperparameters, ”the minimum size of a cluster”, and evaluated the results on the auto2.csv dataset. This hyperparameter determines the minimum number of data points that can be assigned to a cluster. By changing this hyperparameter, we found that the results were comparable to the original ones, suggesting that the hyperparameter optimization is effective in improving the performance of the algorithm. However, further experimentation with other hyperparameters is necessary to determine the optimal values for each hyperparameter to achieve the best performance.

## VII. Conclusion

In this paper, we presented a multi-objective semi-supervised explanation system using the fishing algorithm. We modified the Sway and Xpln functions by replacing the cosine similarity measure with the Jaccard similarity measure. The modified functions produced some different results compared to the original implementation. Although we could not win in every case, the modified functions were able to produce some promising results. Overall, our approach provides a promising direction for further research on multi-objective semi-supervised explanation systems using the fishing algorithm.

While the modifications did not consistently yield better results, the study highlights the potential for enhancing the algorithm by exploring different similarity measures. Overall, the results suggest that the fishing algorithm can be an effective approach for multi-objective semi-supervised explanation systems.

Future works could include further experimentation with different similarity measures or other modifications to the sway and xpln functions in the fishing algorithm. It could also involve exploring the use of different clustering techniques or other machine learning algorithms to improve the performance of the multi-objective semi-supervised explanation system. Additionally, there could be an investigation into the integration of external domain knowledge or additional data sources to enhance the system's predictive capabilities. Finally, the application of the system to other datasets and domains could also be explored to assess its generalizability and effectiveness.

## References

[1] Alok, A.K., Saha, S. and Ekbal, A. A new semi-supervised clustering technique using multi-objective optimization. Appl Intell 43, 633–661 (2015). https://doi.org/10.1007/s10489-015-0656-z

[2] Zahra Ghasemi, Hadi Akbarzadeh Khorshidi, Uwe Aickelin, Multi-objective Semi-supervised clustering for finding predictive clusters, Expert Systems with Applications, Volume 195, 2022, 116551, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2022.116551.

[3] Akbarzadeh Khorshidi, H., Aickelin, U., Haffari, G. et al. Multi-objective semi-supervised clustering to identify health service patterns for injured patients. Health Inf Sci Syst 7, 18 (2019). https://doi.org/10.1007/s13755-019-0080-6