DSGD++: Reducing Uncertainty and Training Time in the DSGD Classifier through a Mass Assignment Function Initialization Technique

Tarkhanyan, A. and Harutyunyan, A.

September 20, 2025

Outline

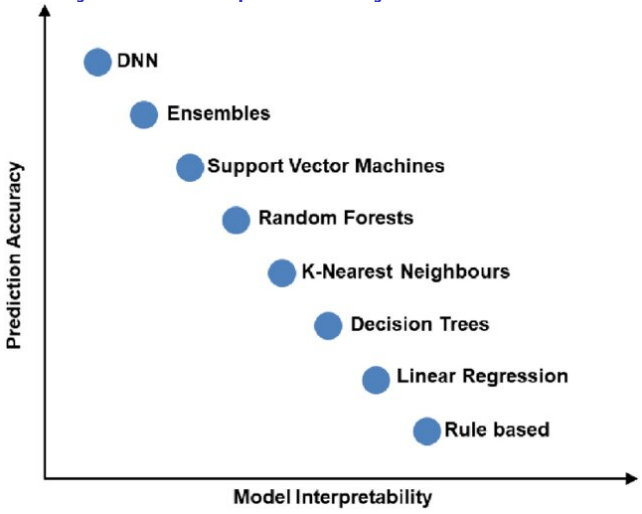
1. Why Interpretable ML?
2. Quick intro to Dempster-Shafer theory (DST)
3. DST + Gradient Descent for Classification
4. Our modifications
5. Results
6. Q&A

Why Interpretable ML?

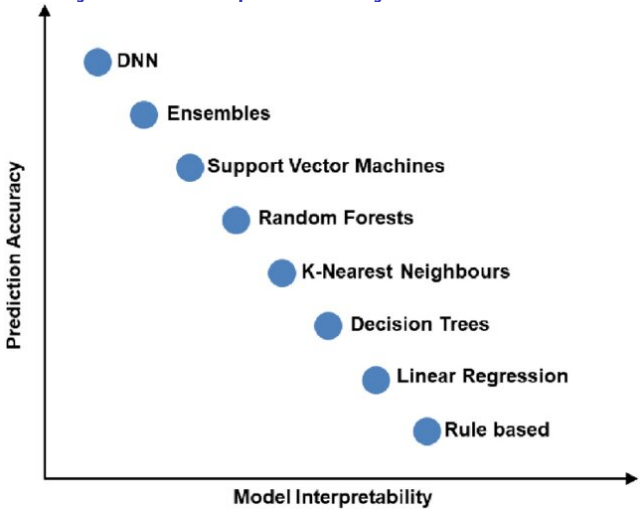
The Rise of Complex Models

- Deep learning and ensemble methods achieve **state-of-the-art performance**
- But they operate as **"black boxes"**
- High accuracy often comes at the cost of interpretability
- Critical applications require **understanding** the decision process:
  - Medical diagnosis
  - Financial lending
  - Legal decisions
  - Autonomous systems

Accuracy vs. Interpretability Tradeoff

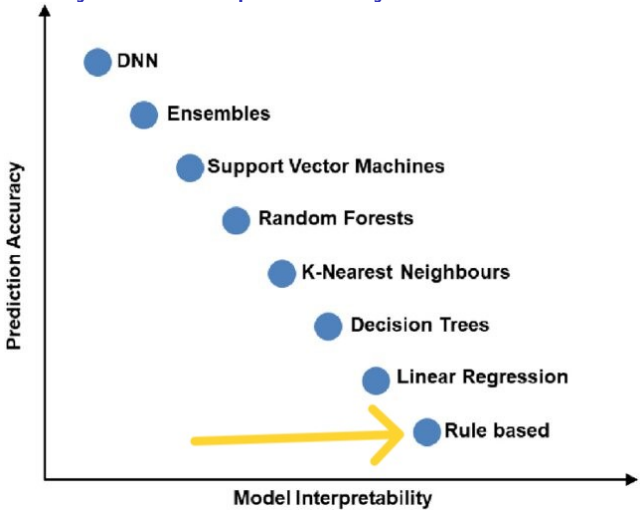


Accuracy vs. Interpretability Tradeoff



- Traditional belief: **more accuracy = less interpretability**
- Our goal: Find methods in the **sweet spot**

Accuracy vs. Interpretability Tradeoff



- Traditional belief: **more accuracy = less interpretability**
- Our goal: Find methods in the **sweet spot**

Regulatory and Ethical Imperatives

GDPR Article 22 - Right to Explanation

"The data subject shall have the right not to be subject to a decision based solely on automated processing... which produces legal effects concerning him or her..."

Regulatory and Ethical Imperatives

GDPR Article 22 - Right to Explanation

"The data subject shall have the right not to be subject to a decision based solely on automated processing... which produces legal effects concerning him or her..."

- **Legal requirement** for explainable AI in EU
- **Algorithmic accountability** - understanding bias and fairness
- **Trust and adoption** - users need to understand system decisions
- **Debugging and improvement** - interpretability helps identify model failures

Why Dempster-Shafer Theory?

- **Inherently interpretable**: Based on simple, human-readable rules
- **Uncertainty quantification**: Explicit modeling of uncertainty
- **Competitive accuracy**: Performance comparable to black-box methods
- **Rule-based explanations**: Easy to understand "if-then" logic

Why Dempster-Shafer Theory?

- **Inherently interpretable**: Based on simple, human-readable rules
- **Uncertainty quantification**: Explicit modeling of uncertainty
- **Competitive accuracy**: Performance comparable to black-box methods
- **Rule-based explanations**: Easy to understand "if-then" logic

Our Contribution

DSGD++ improves training efficiency while maintaining interpretability and accuracy

Dempster-Shafer theory

Dempster-Shafer theory

Dempster-Shafer theory (DST)

General description of DST

DST (also known as "theory of belief functions") provides a mathematical approach for combining evidence from different sources to calculate the probability of an event, utilizing Dempster's rule of combination.

Mass Assignment Function (MAF)

- **Mathematical formulation:**
  - Let X be the set of events, known as the frame of discernment.

Mass Assignment Function (MAF)

- **Mathematical formulation:**
  - Let X be the set of events, known as the frame of discernment.
  - The mass assignment function m is a function defined on the set of subsets of X, 2^X, such that:

$$m : 2^X \rightarrow [0, 1]$$

Mass Assignment Function (MAF)

- **Mathematical formulation:**
  - Let X be the set of events, known as the frame of discernment.
  - The mass assignment function m is a function defined on the set of subsets of X, 2^X, such that:

$$m : 2^X \rightarrow [0, 1]$$

  - The following conditions hold:
    1. $m(\emptyset) = 0$ (The empty set has no mass)
    2. $\sum_{A \subseteq X} m(A) = 1$ (The sum of masses is always 1)

Examples of MAF

- Imagine we are flipping a fair coin. In a classic probability model, this can be expressed as $P(\{heads\}) = P(\{tails\}) = 0.5$. In the DST model, it would be $m(\emptyset) = 0, m(\{heads\}) = m(\{tails\}) = 0.5, m(\{heads, tails\}) = 0$.

Examples of MAF

- Imagine we are flipping a fair coin. In a classic probability model, this can be expressed as $P(\{heads\}) = P(\{tails\}) = 0.5$. In the DST model, it would be $m(\emptyset) = 0, m(\{heads\}) = m(\{tails\}) = 0.5, m(\{heads, tails\}) = 0$.
- If the coin were unfair, with the classic approach we could not assert anything, whereas with DST we can say that $m(\emptyset) = 0, m(\{heads\}) = m(\{tails\}) = 0, m(\{heads, tails\}) = 1$.

Examples of MAF

- Imagine we are flipping a fair coin. In a classic probability model, this can be expressed as $P(\{heads\}) = P(\{tails\}) = 0.5$. In the DST model, it would be $m(\emptyset) = 0, m(\{heads\}) = m(\{tails\}) = 0.5, m(\{heads, tails\}) = 0$.
- If the coin were unfair, with the classic approach we could not assert anything, whereas with DST we can say that $m(\emptyset) = 0, m(\{heads\}) = m(\{tails\}) = 0, m(\{heads, tails\}) = 1$.
- An example where DST can shine is the following. Imagine a person has seen a car passing by and makes the following statement:
  1. The car was either black or brown, in any case, it seemed to be black, but I might be mistaken. In this case, the MAF could look like this: $m(\{\emptyset\}) = 0.1, m(\{black\}) = 0.4, m(\{brown\}) = 0.3, m(\{black, brown\}) = 0.2$

Belief and Plausibility

For all $A \subseteq X$:

$$Bel(A) = \sum_{B \subseteq A} m(B)$$

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B)$$

$$Bel(A) \leq P(A) \leq Pl(A)$$

In the previous example for the black car, we would have:
$m(\{\emptyset\}) = 0.1, m(\{black\}) = 0.4, m(\{brown\}) = 0.3, m(\{black, brown\}) = 0.2$
Let $A =$ black, $B =$ brown

$$Bel(A) = m(A) = 0.4$$

$$Pl(A) = m(A) + m(\{A, B\}) = 0.4 + 0.2 = 0.6$$

Dempster's Rule of Combination

- Dempster's rule is used to combine two mass assignment functions m_1 and m_2 into a new one m_f.
- **Formula:**

$$m_f(A) = m_1(A) \oplus m_2(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C)$$

Dempster's Rule of Combination

- Dempster's rule is used to combine two mass assignment functions m_1 and m_2 into a new one m_f.
- **Formula:**

$$m_f(A) = m_1(A) \oplus m_2(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C)$$

- **Measure of conflict K:**

$$K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$$

Dempster's Rule of Combination

- Dempster's rule is used to combine two mass assignment functions m_1 and m_2 into a new one m_f.
- **Formula:**

$$m_f(A) = m_1(A) \oplus m_2(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C)$$

- **Measure of conflict K:**

$$K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$$

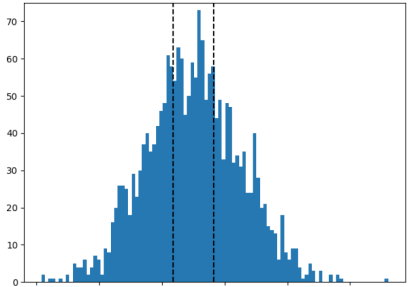
- If $K = 1$, it leads to a division by zero problem, indicating complete conflict between the evidence.

DST + Gradient Descent

DST + Gradient Descent

In order to get the prediction

1. For the given dataset *RS* rule set is generated.
2. Each rule's corresponding *MAF* gets initiated in the following way: Uncertainty (whole set) gets 0.8 weight, and the remaining 0.2 weight is randomly split between *singleton* elements.
3. For given input x:



s(x): x < 0.39	
X	m(X)
A	0.04
B	0.06
A,B	0.90

s(x): 0.39 ≤ x ≤ 1.69	
X	m(X)
A	0.08
B	0.02
A,B	0.90

s(x): x > 1.69	
X	m(X)
A	0.03
B	0.07
A,B	0.90

DST + Gradient Descent

$$\mathcal{M}_x = \{m \mid (m, s) \in RS \wedge x\}$$

$$m_f = \bigoplus_{m \in \mathcal{M}_x} m$$

$$\hat{y} = \underset{\text{class}}{\operatorname{argmax}} \operatorname{Bel}(m_f)$$

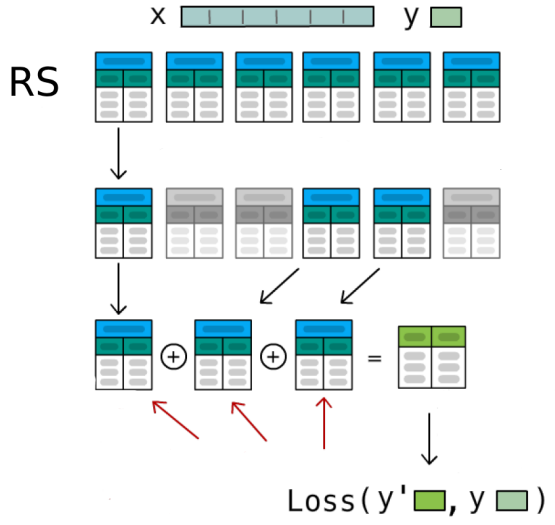
Loss function

Mean squared error

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \|y_i - \hat{y}_i\|^2 \tag{1}$$

Cross-Entropy

$$CE = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} y_{ij} \cdot \log(\hat{y}_{ij}) \tag{2}$$


x
y
RS
Loss(y', y)

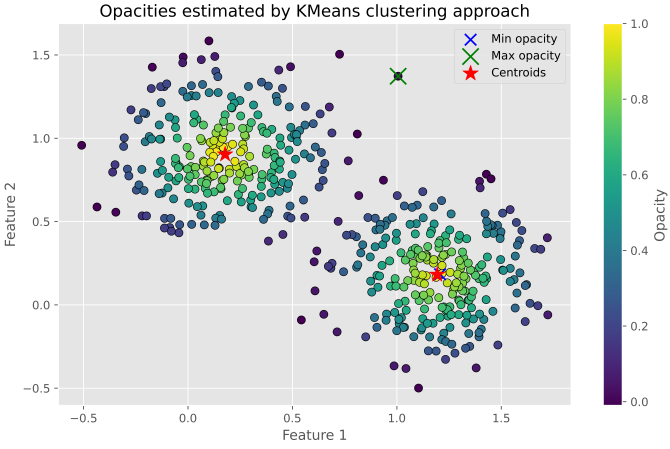
DSGD++: DSGD with better initialization

Algorithm 1 Opacity Estimation using KMeans Clustering

Require: X: Dataset, k: Number of clusters (same as dataset's number of classes),
Ensure: $opacityList$: List of opacity values for each data point

function GET_CONFIDENCE_KMEANS(X, k)
    Standardize the dataset X using standard scaling.
    Fit the KMeans clustering algorithm on X with k clusters to obtain centroids $\{C_i\}_{i=0}^{k-1}$.
    $confidenceList \leftarrow$ Empty list
    for each $dataPoint$ in X **do**
        Calculate Euclidean distance from $dataPoint$ to its nearest centroid C_i.
    end for
    Identify outliers among the data points using the Z-score technique, where a data point is considered an outlier if its distance's Z-score exceeds $zScoreThrehold$.
    for each $dataPoint$ in X **do**
        if $dataPoint$ is not an outlier **then**
            $normalizedOpacity \leftarrow dataPoint$'s distance after normalizing using min-max scaling within its cluster.
            $opacity \leftarrow 1 - normalizedOpacity$
        else
            $opacity \leftarrow 0$
            Append $opacity$ to $opacityList$
        end if
    end for
    return $opacityList$
end function

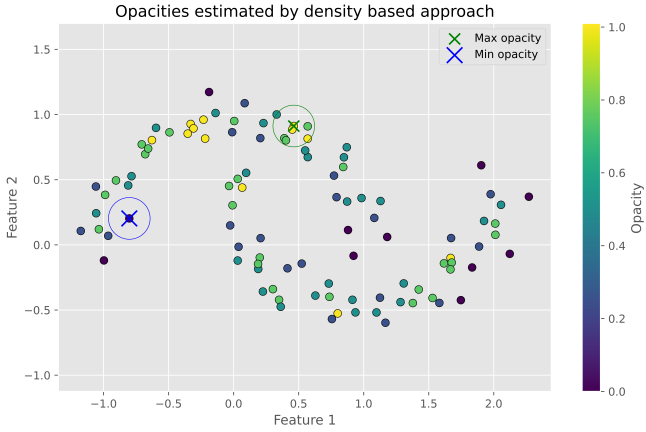
Figure: KMeans Algorithm


Opacities estimated by KMeans clustering approach
Min opacity
Max opacity
Centroids
Feature 1
Feature 2
Opacity
1.5
1.0
0.5
0.0
−0.5
−0.5
0.0
0.5
1.0
1.5
1.0
0.8
0.6
0.4
0.2
0.0

Algorithm 2 Opacity Estimation using Density Based Approach

Require: X: Dataset, k: Number of clusters (same as number of unique classes),
Ensure: $confidenceList$: List of confidence scores for each data point

function GET_CONFIDENCE_DBSCAN(X, k)
    Standardize the dataset X using standard scaling.
    Initialize $eps \leftarrow$ Initial small value (e.g., 0.1)
    while $numCentroids \neq k$ and $eps \leq maxEps$ **do**
        $model \leftarrow$ DBSCAN(eps, $minPoints$)
        Fit DBSCAN on X
        $numCentroids \leftarrow$ Count of unique clusters formed (excluding noise)
        $eps \leftarrow eps + step$
    end while
    $radius \leftarrow eps$
    Initialize $confidenceList \leftarrow$ Empty list
    for each $dataPoint$ in X **do**
        $opacity \leftarrow \sum_{y \in X} \mathbf{1}_{\|y-x\| \leq radius}$
        Append $opacity$ to $opacityList$
    end for
    $opacityList \leftarrow$ Min-Max Scaled version of $opacityList$
    return $opacityList$
end function


Opacities estimated by density based approach
Max opacity
Min opacity
Feature 1
Feature 2
Opacity
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
1.0
0.8
0.6
0.4
0.2
0.0

Rule confidence estimation

1. Filter the dataset to retain only the rows that comply with the rule.

Rule confidence estimation

1. Filter the dataset to retain only the rows that comply with the rule.
2. If the rule does not apply to any rows, set its confidence to 0 (this corresponds to the full uncertainty). Otherwise:

Rule confidence estimation

1. Filter the dataset to retain only the rows that comply with the rule.
2. If the rule does not apply to any rows, set its confidence to 0 (this corresponds to the full uncertainty). Otherwise:
3. Calculate the rule's confidence as the mean representativeness of the rows it covers.

Rule confidence estimation

1. Filter the dataset to retain only the rows that comply with the rule.
2. If the rule does not apply to any rows, set its confidence to 0 (this corresponds to the full uncertainty). Otherwise:
3. Calculate the rule's confidence as the mean representativeness of the rows it covers.
4. If the rows are not homogeneous with respect to their labels, reduce the confidence based on the proportion of the most frequent label among these rows.

MAF Initialization

$$m(l_i) = \begin{cases} c & \text{if } l_i = l_{\text{mode}}, \\ \frac{1-c}{n-1} & \text{otherwise,} \end{cases}$$

Results

Data

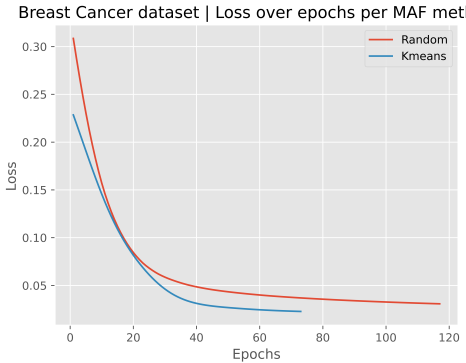
Dataset	Rows	Columns	Description
Brain Tumor	3762	14	Includes first-order and texture features with target levels.
Breast Cancer Wisconsin	699	9	Clinical reports detailing cell benignity or malignancy.
Gaussian	500	3	Two 2D Gaussian distributions generate this dataset.
Uniform	500	3	Uniform samples from [-5, 5], with class split by the sign of x.
Rectangle	1263	3	Points in [-1, 1]×[-1, 1], class determined by the y component's sign.

Accuracy and Speedup Analysis

MAF method	dataset	accuracy	f1	training_time	epochs	min_loss	initial_loss
clustering	Brain Tumor	0.981	0.98	138.469	117	0.018	0.181
random	Brain Tumor	0.983	0.981	157.075	132	0.026	0.245
clustering	Breast Cancer	0.976	0.966	17.243	73	0.023	0.228
random	Breast Cancer	0.976	0.966	19.63	117	0.031	0.308
clustering	Gaussian	0.987	0.988	15.792	100	0.017	0.083
random	Gaussian	0.987	0.988	39.059	264	0.024	0.265
clustering	Rectangle	1	1	61.032	167	0.006	0.252
random	Rectangle	1	1	98.64	275	0.008	0.235
clustering	Uniform	0.973	0.97	20.683	120	0.035	0.197
random	Uniform	0.973	0.97	39.208	273	0.037	0.255

dataset	accuracy_ratio	f1_ratio	time_speedup, x
Brain Tumor	1	1	1.13
Breast Cancer	1	1	1.14
Gaussian	1	1	2.47
Rectangle	1	1	1.62
Uniform	1	1	1.90

Average speedup - 1.65x



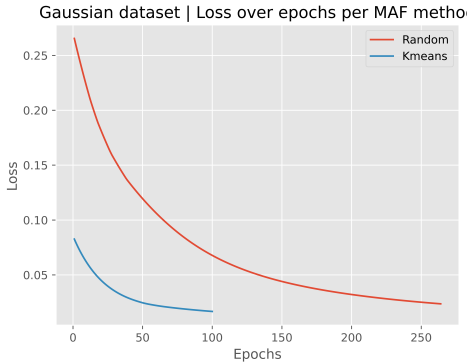


Figure: The figure demonstrates the benefits offered by MAF initialization, mainly decreased number of epochs and better starting point.

Pitfall of the current approach

Rule	Mass First Class	Mass Second Class	Uncertainty	Ratio
1	0.49	0.51	0	1.04
2	0.01	0.09	0.9	9

Table 2. Illustration of the Pitfalls of the Traditional Approach

- **Uncertainty Adjustment:**

$$U' = 1 - U$$

where U is the original uncertainty.

- **Uncertainty Adjustment:**

$$U' = 1 - U$$

where U is the original uncertainty.

- **Normalization of the Ratio:**

$$R' = \frac{R - \min(R)}{\max(R) - \min(R)}$$

where R is the original ratio (when dividing the values of two masses we add $\varepsilon = 0.01$ to denominator to avoid zero division error), and $\min(R)$ and $\max(R)$ are the minimum and maximum values of the ratio across all rules, respectively.

- **Uncertainty Adjustment:**

$$U' = 1 - U$$

where U is the original uncertainty.

- **Normalization of the Ratio:**

$$R' = \frac{R - \min(R)}{\max(R) - \min(R)}$$

where R is the original ratio (when dividing the values of two masses we add $\varepsilon = 0.01$ to denominator to avoid zero division error), and $\min(R)$ and $\max(R)$ are the minimum and maximum values of the ratio across all rules, respectively.

- **Harmonic Mean Calculation:**

$$H = \frac{2 \cdot U' \cdot R'}{U' + R'}$$

Dataset	mean_clustering	mean_random	median_clustering	median_random	improvement factor
Brain Tumor	0.258	0.713	0.246	0.732	2.979
Breast Cancer	0.314	0.710	0.262	0.703	2.685
Gaussian	0.225	0.330	0.229	0.302	1.318
Rectangle	0.201	0.448	0.203	0.434	2.135
Uniform	0.150	0.262	0.096	0.144	1.493

improvement_factor is defined as the ratio of *median_random* and *median_clustering*.
On average the clustering approach yields in uncertainty reduction by a factor of **2.12**.

Dataset	mean_clustering	mean_random	median_clustering	median_random	improvement factor
Brain Tumor	0.247	0.313	0.075	0.336	4.471
Breast Cancer	0.156	0.359	0.075	0.411	5.486
Gaussian	0.633	0.625	0.807	0.757	0.937
Rectangle	0.473	0.536	0.457	0.571	1.249
Uniform	0.732	0.700	0.935	0.876	0.937

On average the clustering approach yields in uncertainty reduction (harmonic mean approach) by a factor of **2.61**.

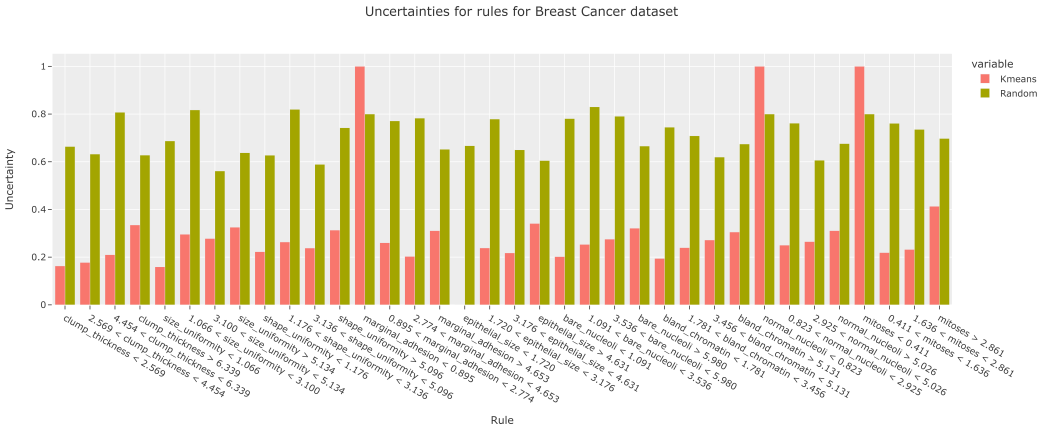


Figure: Uncertainties per rule for different MAF initialization methods

Promising Next Steps

- Extend to multi-class classification problems
- Validate on larger, more complex datasets
- Combine the approach with classical rule induction algorithms (e.g., RIPPER, C4.5, FOIL)

Bibliography (1/3)

- Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
- Peñafiel, S., Baloian, N., Sanson, H., & Pino, J. A.: Applying Dempster–Shafer theory for developing a flexible, accurate and interpretable classifier. Expert Systems with Applications **148**, 113262 (2020)
- MacQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. Proc. 5th Berkeley Symp. on Math. Statist. and Prob., Vol. 1 (1967), 281–297
- Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD'96), AAAI Press (1996), 226–231

Bibliography (2/3)

- Baloian, N., Davtyan, E., Petrosyan, K., Poghosyan, A., Harutyunyan, A., Penafiel, S.: Embedded Interpretable Regression using Dempster-Shafer Theory. Proceedings of the 4th Codassca Workshop on Data Science and Reliable Machine Learning (2024)
- Baloyan, A., Aramyan, A., Baloian, N., Poghosyan, A., Harutyunyan, A., Penafiel, S.: An Empirical Analysis of Feature Engineering for Dempster-Shafer Classifier as a Rule Validator. Proceedings of the 4th Codassca Workshop on Data Science and Reliable Machine Learning (2024)
- Valdivia, R., Baloian, N., Chahverdian, M., Adamyan, A., Harutyunyan, A.: An Explainable Clustering Algorithm using Dempster-Shafer Theory. Proceedings of the 4th Codassca Workshop on Data Science and Reliable Machine Learning (2024)

Bibliography (3/3)

- Wolberg, W. H., Mangasarian, O. L.: Multisurface method of pattern separation for medical diagnosis applied to breast cytology. Proceedings of the National Academy of Sciences **87**(23), 9193–9196 (1990)
- Bohaju, J.: Brain Tumor. In: Kaggle 2020, DOI: . `https://www.kaggle.com/dsv/1370629` (2020)
- Cohen, W.W.: Fast Effective Rule Induction. In: Prieditis, A., Russell, S. (eds.) Machine Learning Proceedings 1995, Morgan Kaufmann (1995), 115–123
- Sedláček, O., Bartoš, V.: Fusing Heterogeneous Data for Network Asset Classification – A Two-layer Approach. In: 2024 IEEE Network Operations and Management Symposium (NOMS) (2024)

Paper and Code

Published Paper

DSGD++: Reducing Uncertainty and Training Time in the DSGD Classifier through a Mass Assignment Function Initialization Technique

Journal of Universal Computer Science

DOI: 10.3897/jucs.164745

Source Code

Open Source Implementation

GitHub Repository:
`https://github.com/HaykTarkhanyan/DSGD-Enhanced`

Complete implementation with datasets and experiments

Thank you