Distance Metric Recommendation for *k*-Means Clustering: A Meta-Learning Approach

Mark Edward M. Gonzales, Lorene C. Uy, Jacob Adrianne L. Sy & Macario O. Cordel, II
{mark_gonzales, lorene_c_uy, jacob_adrianne_l_sy, macario.cordel}@dlsu.edu.ph
De La Salle University, Manila, Philippines

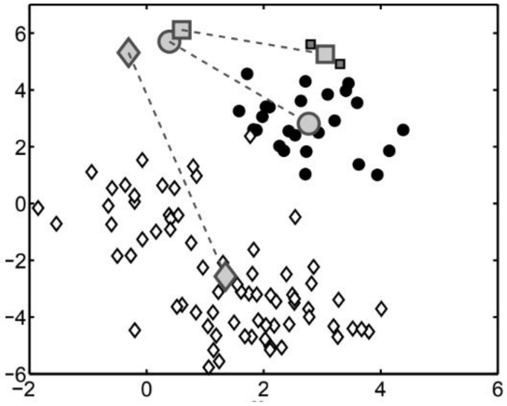
Project Page

Distance Metric and Clustering

- In **centroid-based clustering algorithms**, the distance metric is used to determine the **cluster assignment** of a data point

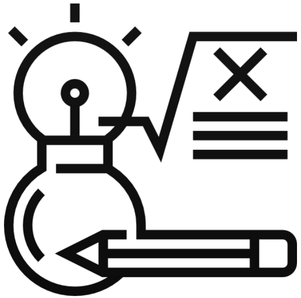
- The impact of the choice of the distance metric on the clustering quality has been empirically demonstrated in several studies

(Suarez, Garcia & Herrera, 2021; Quaddoura et al., 2020; Xing, Ng, Jordan & Russell, 2002)



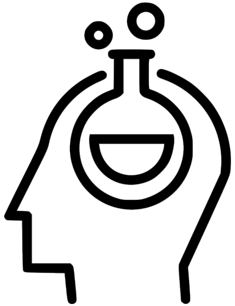
GitHub

Traditional Approaches to Distance Metric Selection



Theoretical

Requires deep expertise
on the geometry of the dataset



Experimental

Demands a significant amount
of time and resources

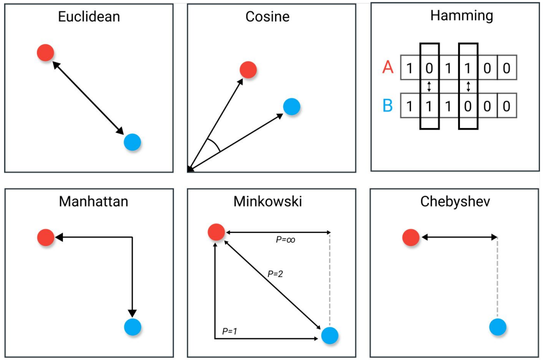
GitHub

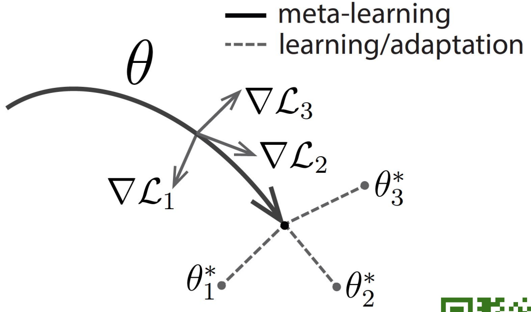
Meta-Learning

- "Learning to learn"

- A subfield of machine learning that explores the **automatic recommendation** of parameters and algorithms, as well as the improvement of their performance

(Lemke, Budka & Gabrys, 2013)





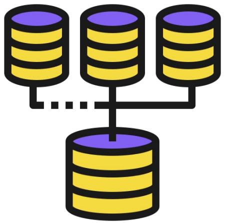
GitHub

Contributions

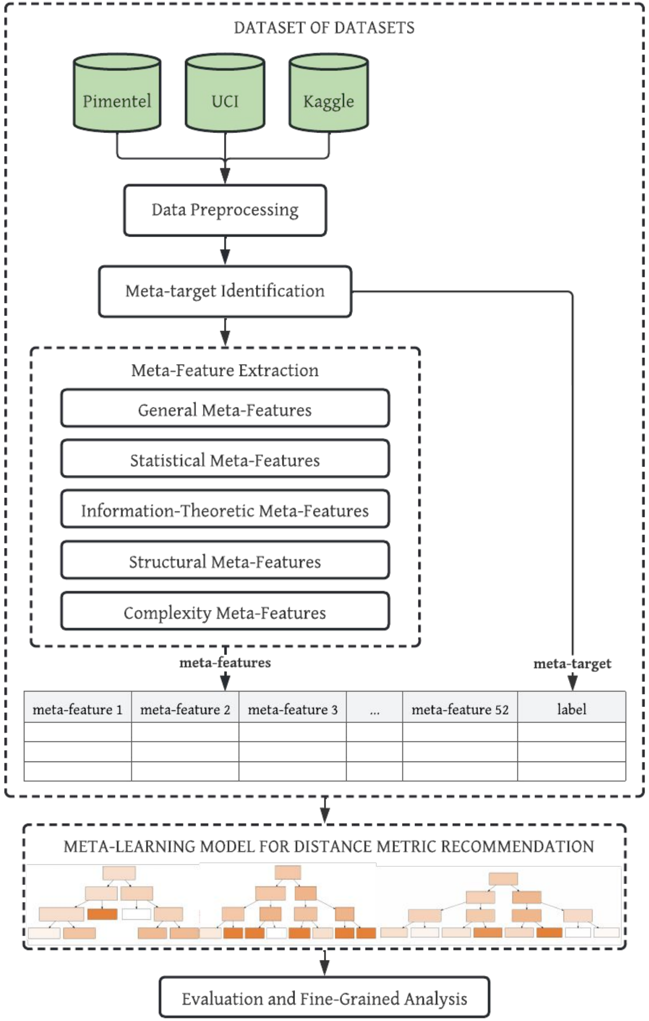
- **Dataset of datasets** for meta-learning studies on clustering
- **Meta-learning model** for distance metric recommendation for *k*-means clustering using (1) general, (2) statistical, (3) information-theoretic, (4) structural, and (5) complexity meta-features
- **Fine-grained analysis** of meta-feature importance and effects

GitHub

Dataset of Datasets



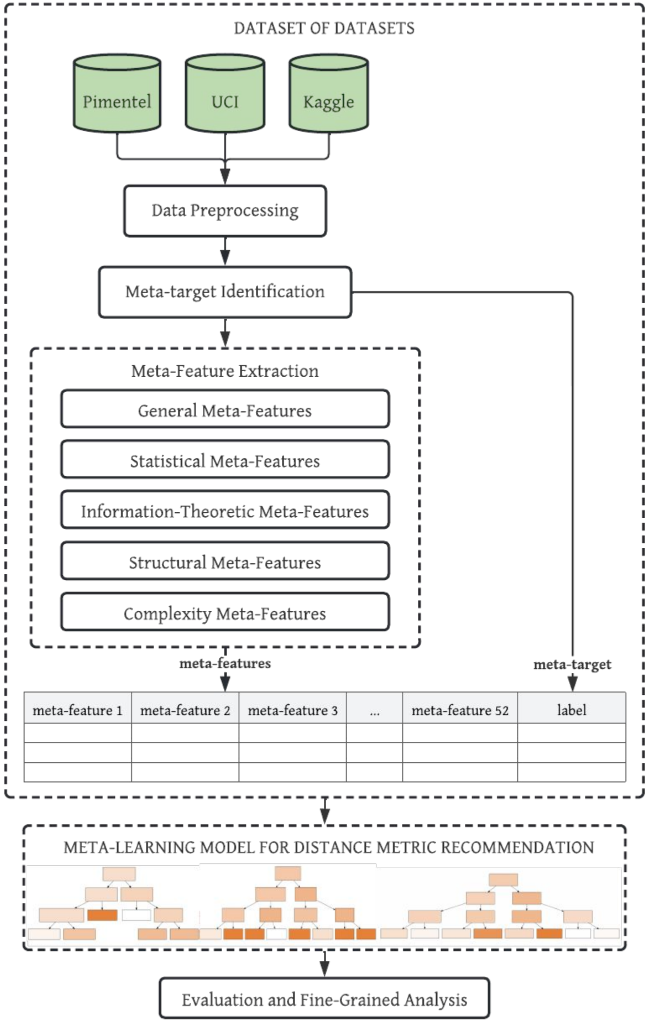
GitHub



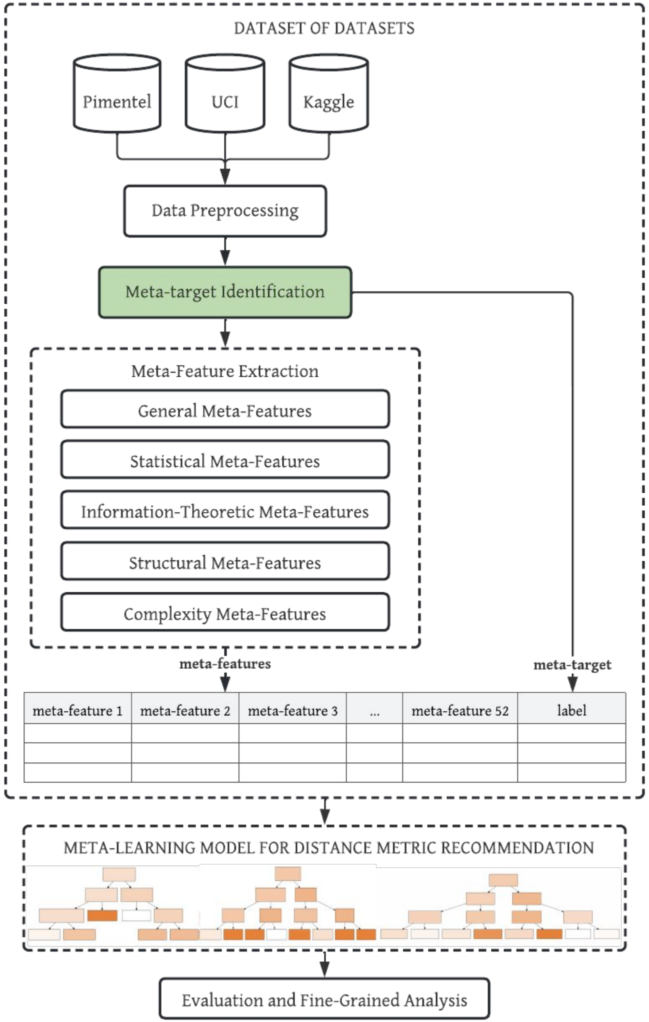
Data Collection

- **340 datasets**
  - 195 from OpenML
  - 60 from UCI
  - 85 from Kaggle

GitHub



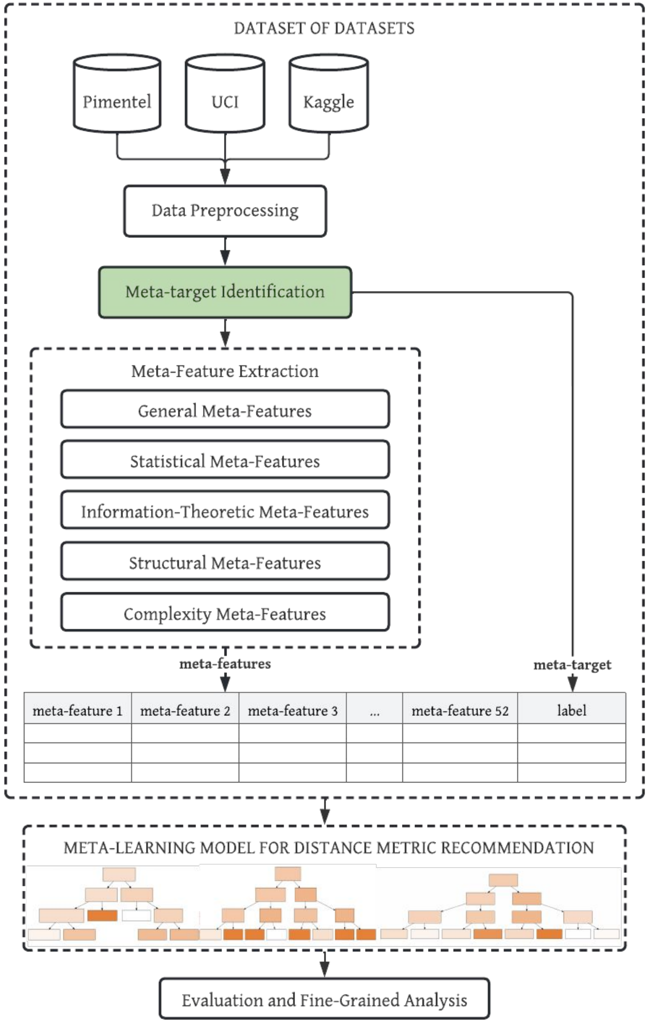
Study	Num. of Dataset Entries	Num. of Meta-Features	Meta-Target
Pimentel & De Carvalho (2019)	57	46	Algorithm
Jilling & Alvarez (2020)	135	25	Algorithm
Pimentel & De Carvalho (2018)	219	19	Algorithm
Muravyov et al. (2017)	200	19	Validation Index
Pimentel & De Carvalho (2020)	219	145	Num. of Clusters
Zhu et al. (2020)	199	41	Distance Metric
Ours	**340**	**52**	**Distance Metric**



Meta-Target Identification

- *k*-means clustering, coupled with a grid search over selected distance measures, was performed to label each dataset in the collection with the distance metric that optimizes the **Davies-Bouldin score**

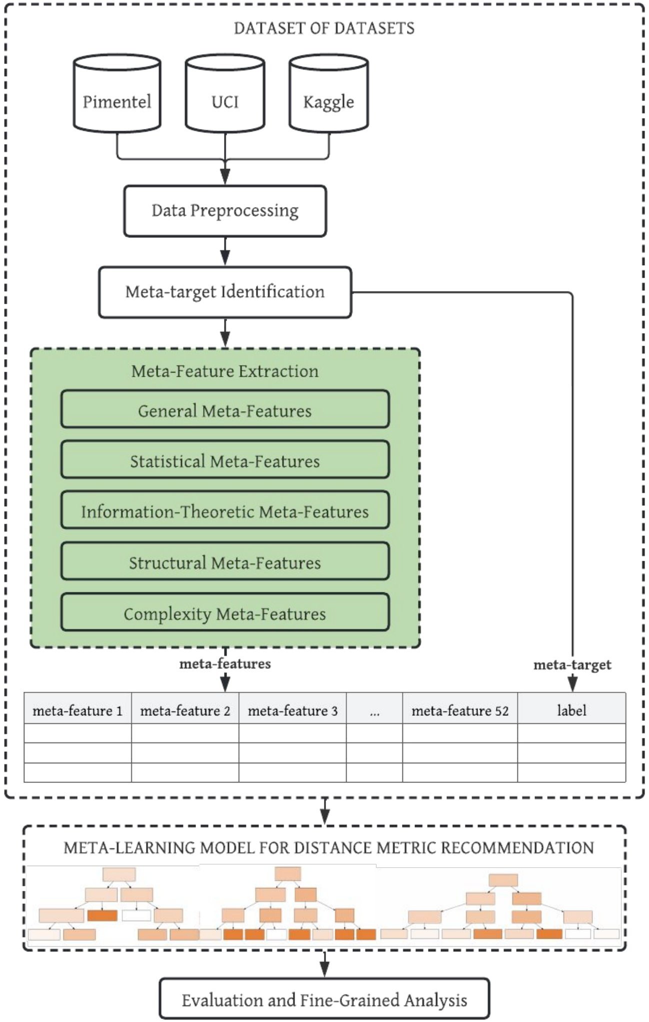
$$\frac{1}{k}\sum_i \max_{j,j\neq i}\left\{\frac{\frac{1}{n_i}\sum_{x\in C_i} d(x,c_i)+\frac{1}{n_j}\sum_{x\in C_j} d(x,c_j)}{d(c_i,c_j)}\right\}$$



Meta-Target Identification

Distance Metric	After Relabeling
Chebyshev	139 (40.88%)
Euclidean	122 (35.89%)
Manhattan	79 (23.23%)

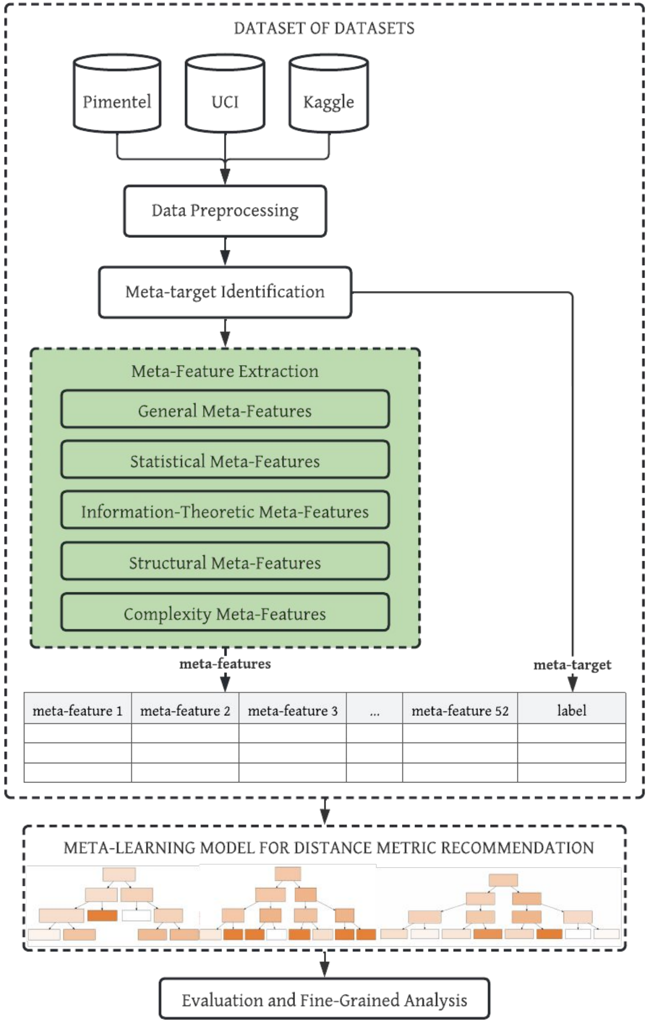
GitHub



Meta-Feature Extraction

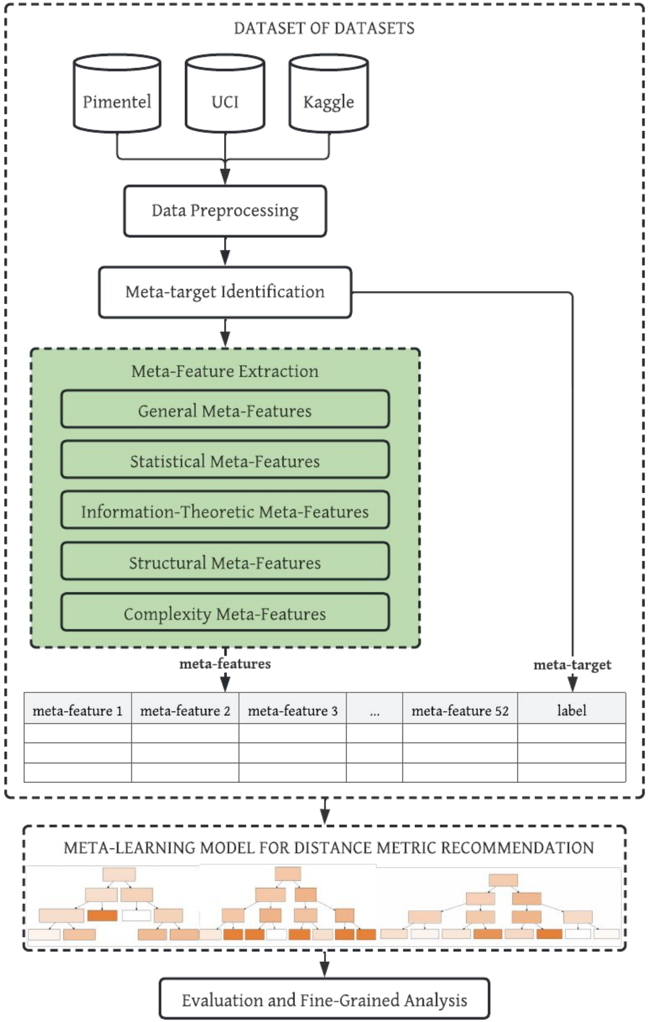
- **52 meta-features**
    - Combined from the works of Zhu et al. (2020), Vanschoren (2019), and Alcobaça et al. (2020)
    - Selected based on the applicability to unsupervised tasks

GitHub



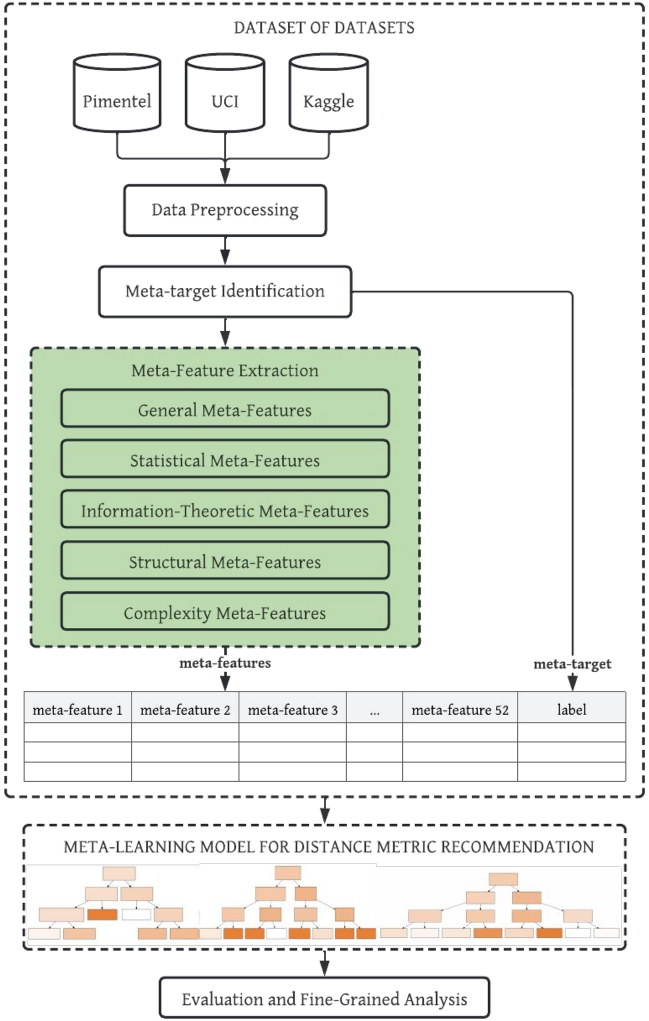
Meta-Feature Extraction

- **General**
  - Describe the dimensionality and size of the dataset (Vanschoren, 2019)
- **Statistical**
  - Capture characteristics related to feature interdependence, normality, degree of discreteness, and noisiness (Vanschoren, 2019)



Meta-Feature Extraction

- **Information-Theoretic**
  - Quantify feature informativeness and interdependence (Vanschoren, 2019; Castiello et al., 2005)
- **Structural**
  - Capture patterns and correlation information from the *k*-itemsets frequencies (Song et al., 2012)

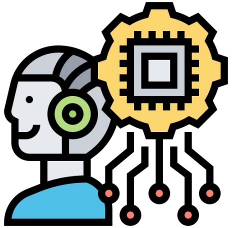


Meta-Feature Extraction

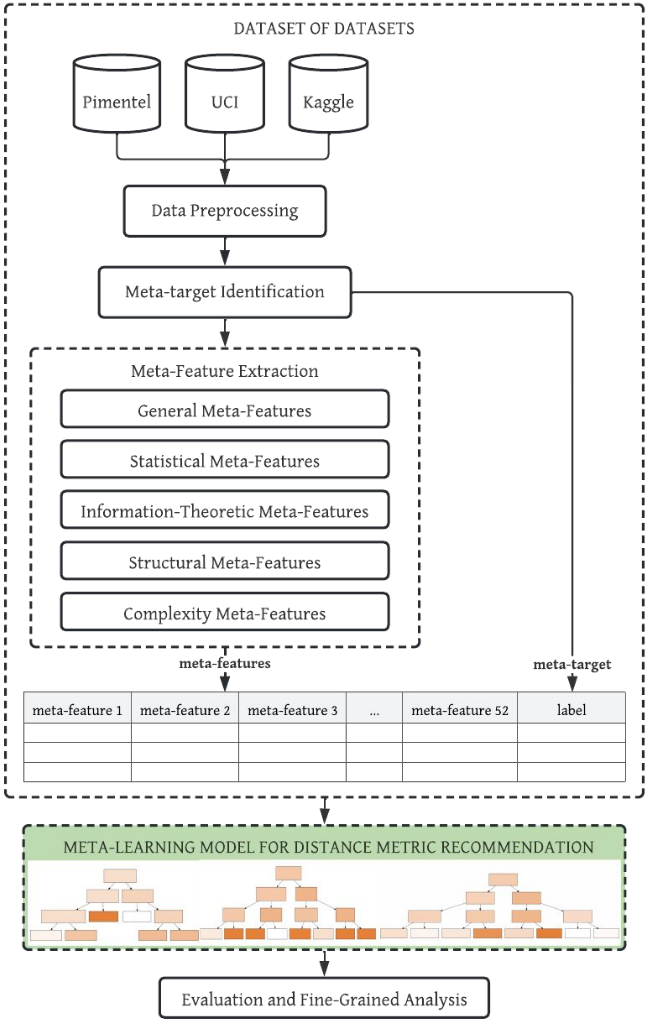
- **Complexity**
  - Pertain to attributes related to the PCA dimensions (Lorena et al., 2019)

GitHub

Meta-Learning Model



GitHub

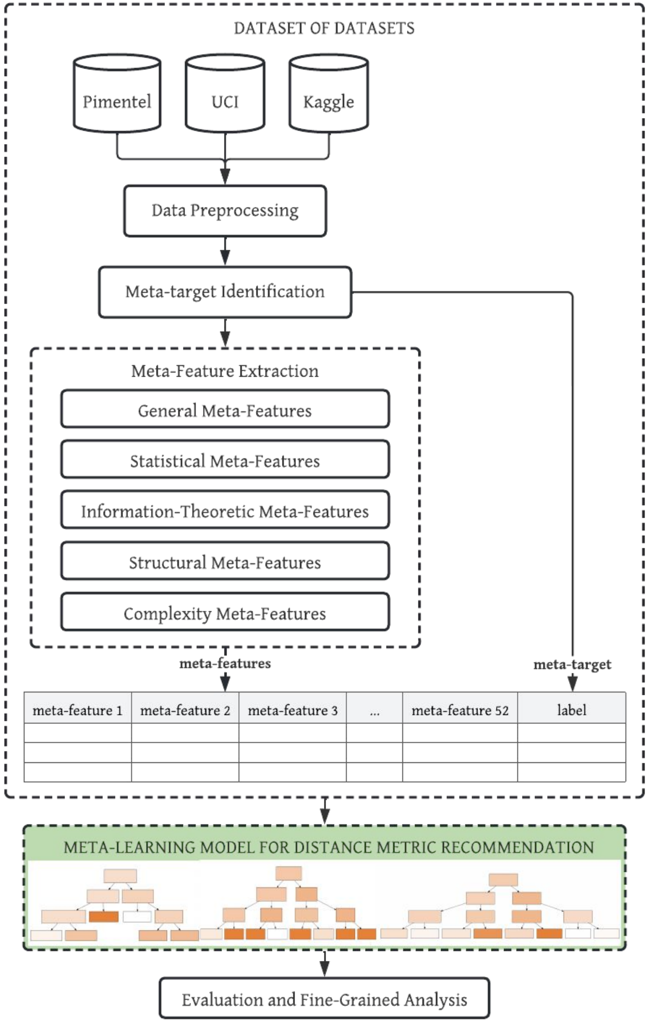


Meta-Learning Model

- **Input:** Vector of meta-features
- **Output:** Distance metric

Training (70%)	Test (30%)

- **Random forest**
  - Bagging makes it robust to noise and overfitting (Fawagreh et al., 2014)
  - Known to perform well on small datasets (Ibrahim & Carman, 2014)



Meta-Learning Model

- **Hyperparameter tuning**
  - Grid search
  - Five-fold stratified cross-validation
  - Maximize accuracy (micro-F1)
- **Addressing class imbalance**
  - SMOTE (Chawla et al., 2002)
  - Borderline SMOTE (Han et al., 2005)
  - ADASYN (He et al., 2008)

Model Evaluation

GitHub

	SMOTE	Borderline SMOTE	ADASYN
Accuracy (Micro-F1)	63.73%	**70.59%**	65.69%
Macro-F1	60.29%	**67.86%**	63.01%
Macro-Precision	60.78%	**67.95%**	63.06%
Macro-Recall	60.32%	**67.92%**	63.10%

Number of trees: 50

Splitting criterion: Gini

Maximum depth: 15

Minimum number of samples to be a leaf node: 3

Minimum number of samples to split an internal node: 2

Number of features to consider at each split: $\log_2$ of the number of features

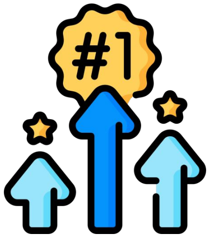
Warm start: True

Minimum impurity decrease: 0.0

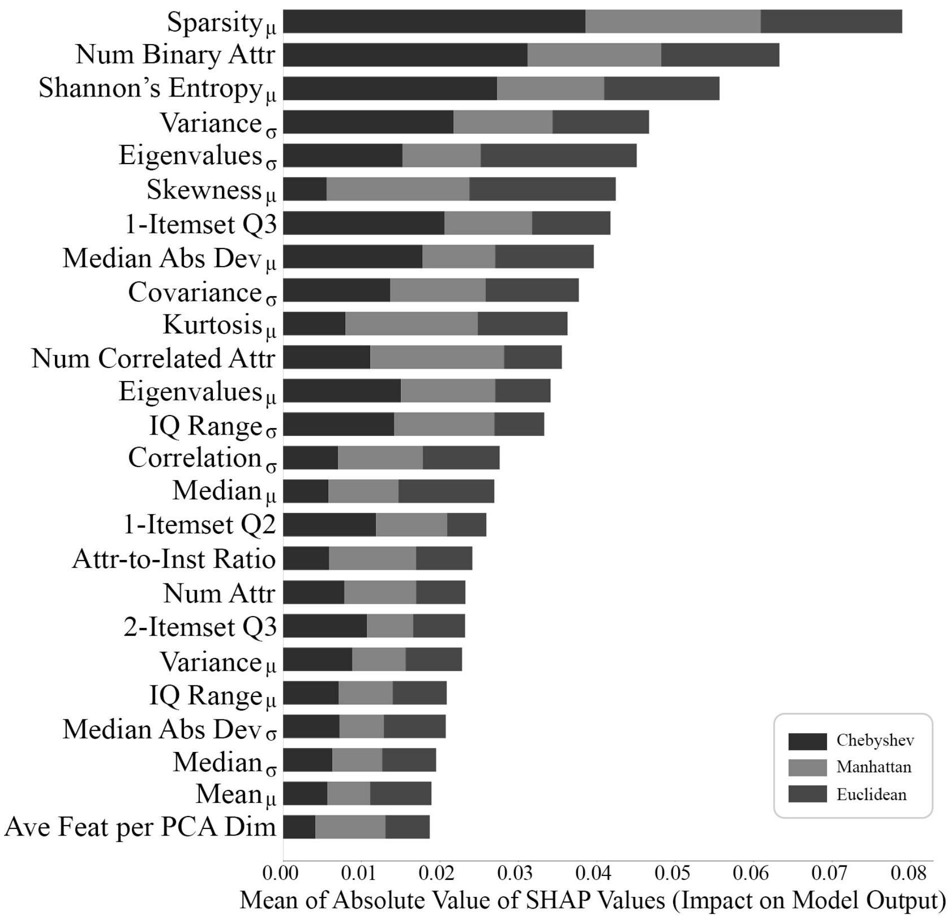
Complexity parameter *α* for minimal cost-complexity pruning: 0.0

GitHub

Feature Importance



GitHub



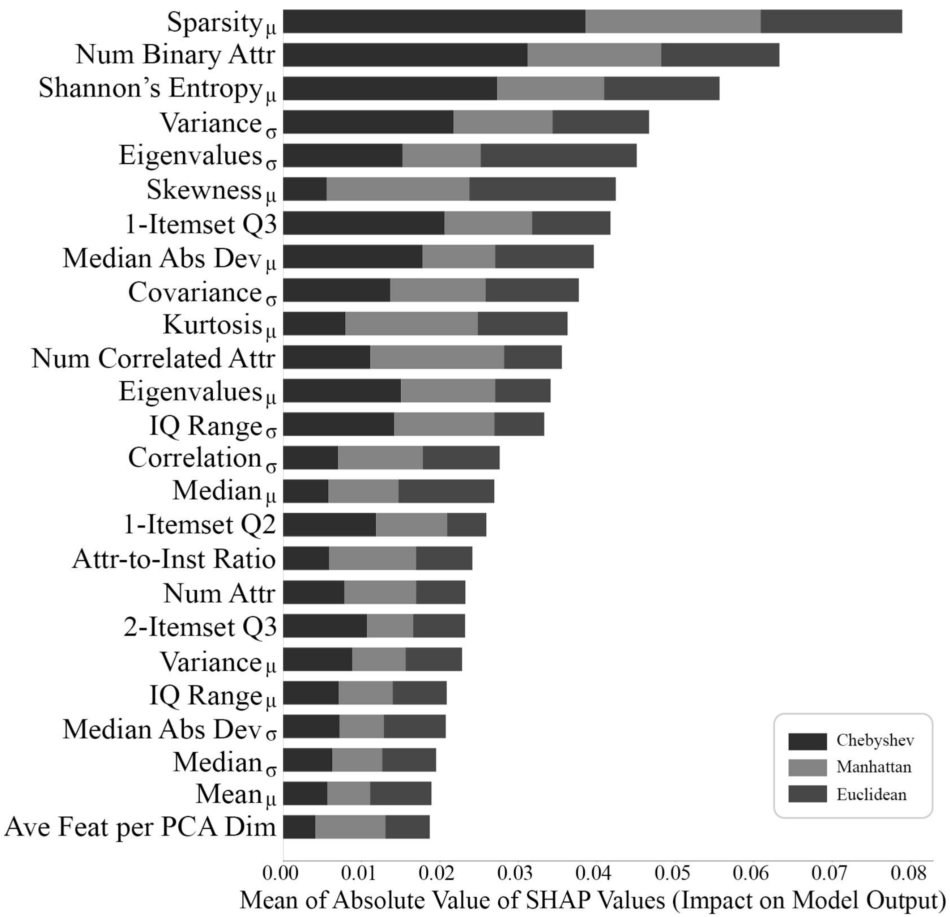
Global Feature Importance

- Average of the absolute values of the SHAP value per feature across the dataset

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z_j'$$

$$I_j = \frac{1}{N} \sum_{j=1}^{N} | \phi_j^{(i)} |$$

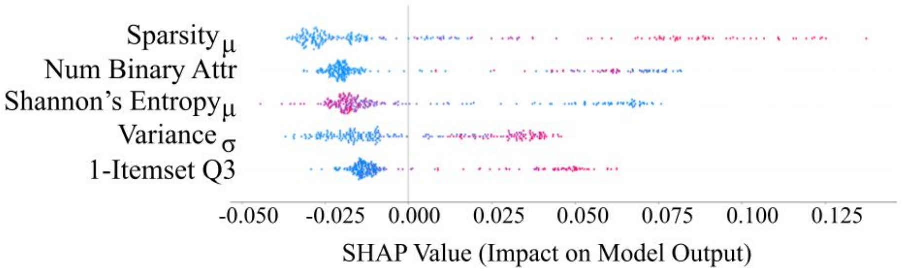
GitHub



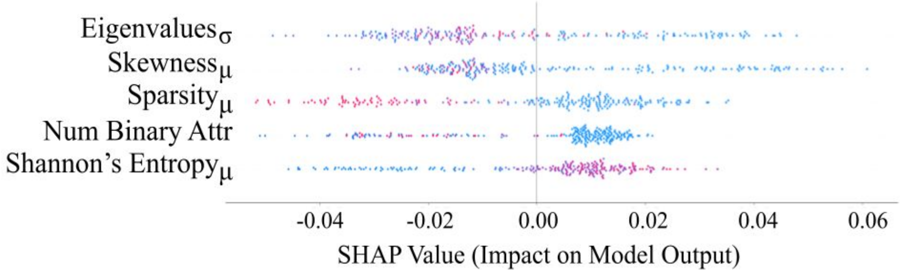
Global Feature Importance

- **Top 5 Meta-Features**
  - Sparsity_{μ}
  - Number of Binary Attributes
  - $\text{Shannon's Entropy}_{\mu}$
  - Variance_{σ}
  - $\text{Eigenvalues}_{\sigma}$

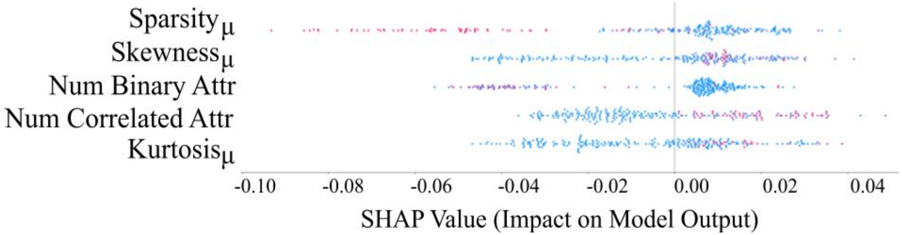
- These meta-features, except $\text{Shannon's entropy}_{\mu}$, have not been considered in prior studies



Chebyshev Distance



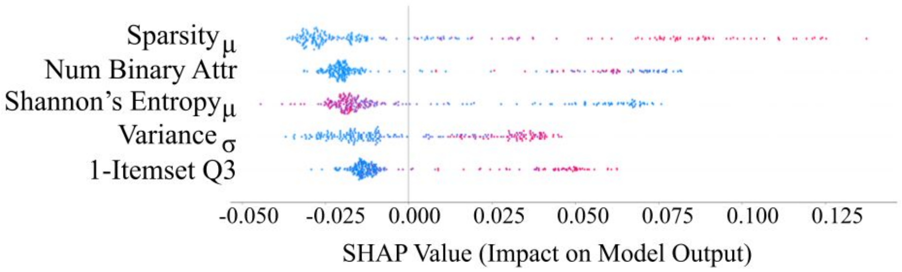
Euclidean Distance



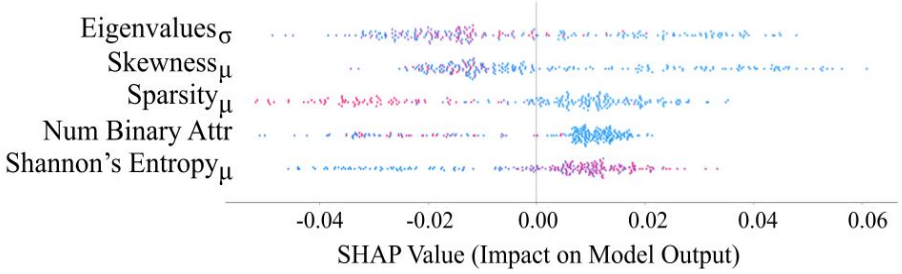
Manhattan Distance

Aside from their high global contribution, the sparsity$_\mu$ and the number of binary attributes are consistently among the top five meta-features with the highest importance relative to each of the three distance measures

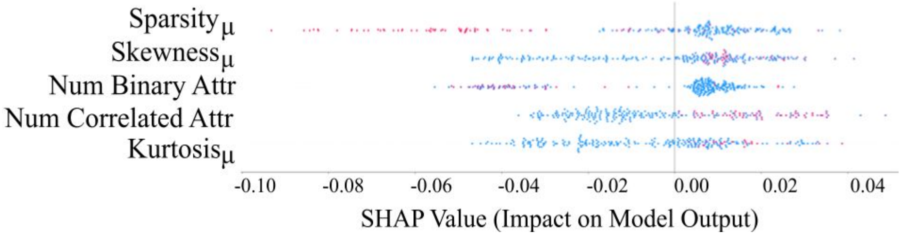
GitHub



Chebyshev Distance



Euclidean Distance



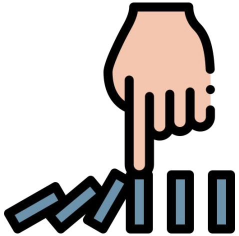
Manhattan Distance

The sparsity is a measure of discreteness. Let *n* be the number of instances in the dataset and $\phi(a)$ be the number of distinct values under attribute a

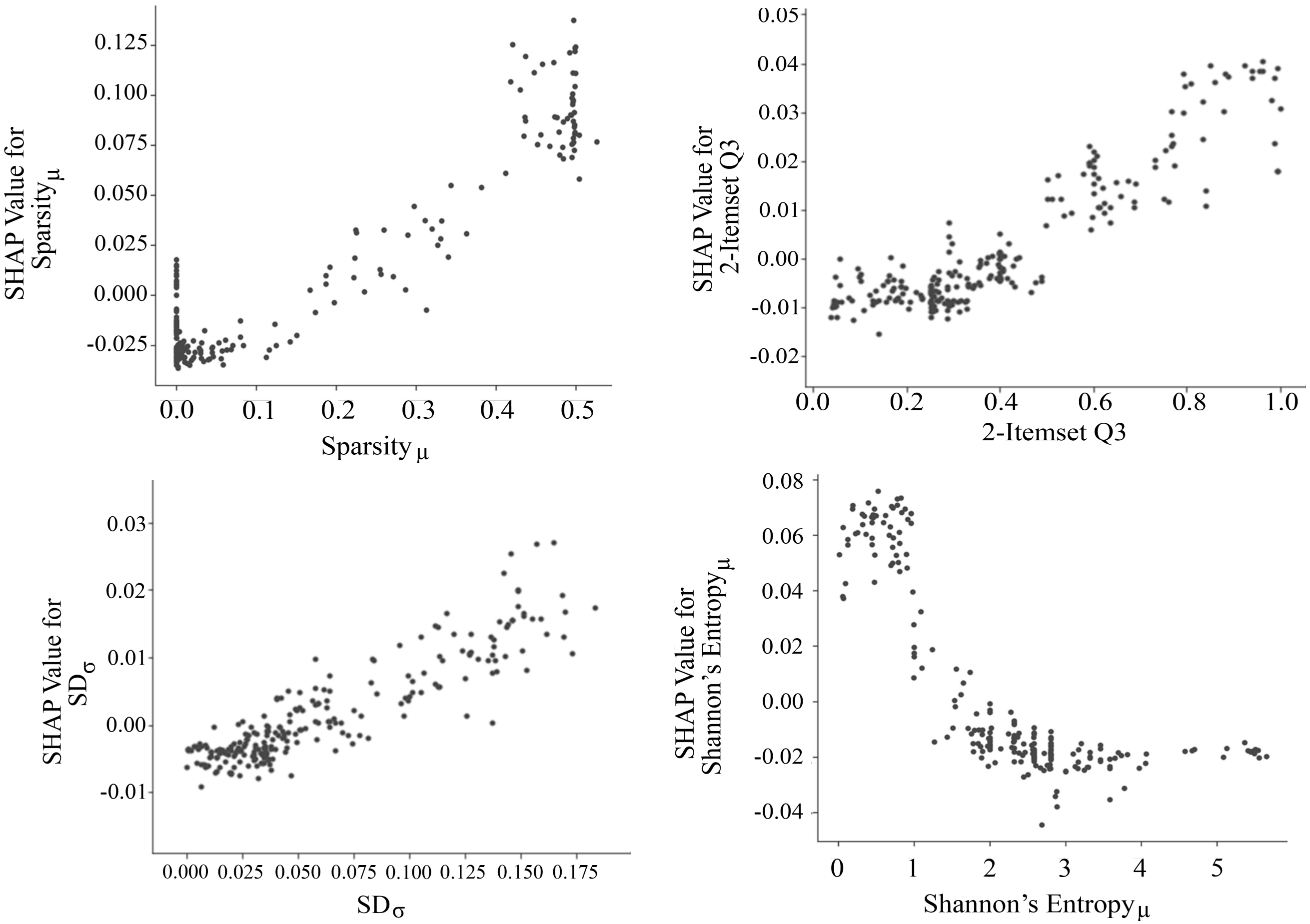
$$\frac{1}{n-1}\left(\frac{n}{\phi(a)}-1\right)$$

GitHub

Feature Effects

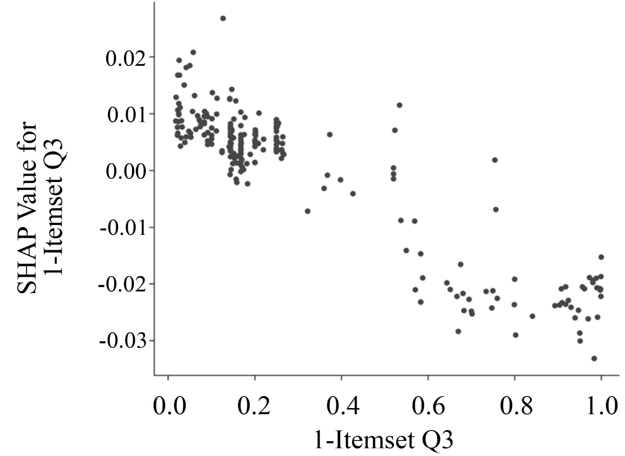


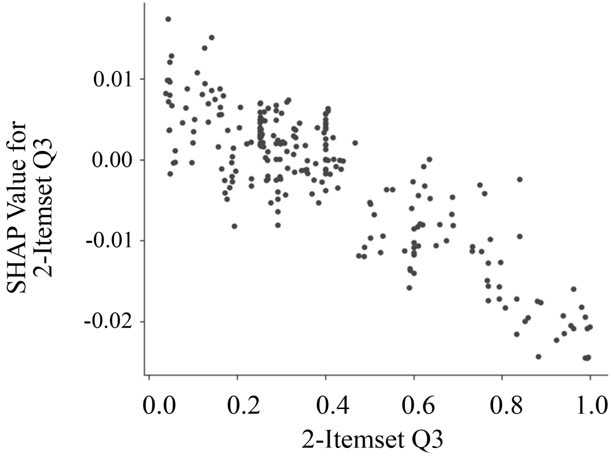
GitHub



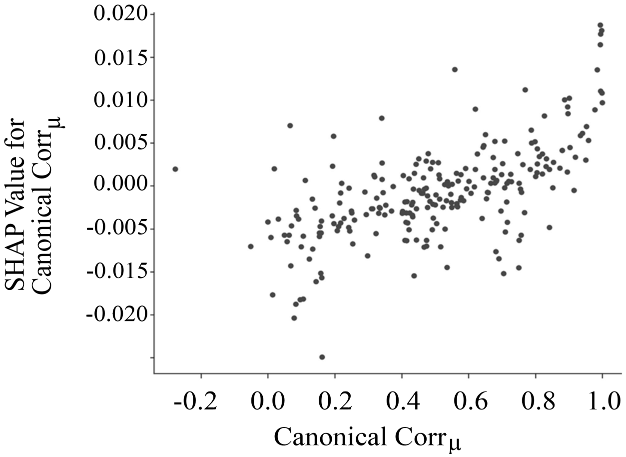
Chebyshev Distance

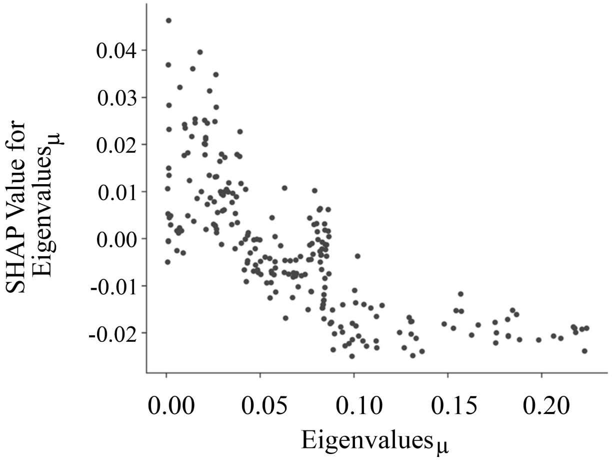
GitHub





Euclidean Distance

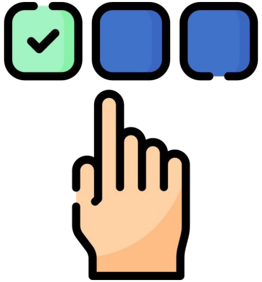




Manhattan Distance

GitHub

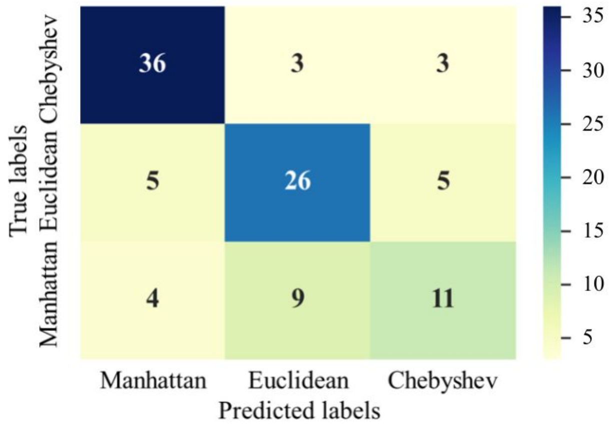
Feature Selection



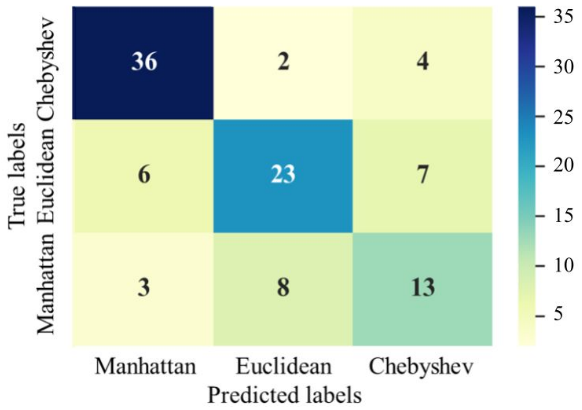
GitHub

	All 52 Meta-Features	Top 25 Meta-Features
Accuracy (Micro-F1)	70.59%	**71.57%**
Macro-F1	67.86%	**68.06%**
Macro-Precision	67.95%	**68.77%**
Macro-Recall	**67.92%**	**67.92%**

GitHub



Top 25 Meta-Features



All 52 Meta-Features

Misclassifications

- Most misclassifications were instances under Manhattan distance that were incorrectly classified under Euclidean distance

- While borderline SMOTE was applied in an attempt to address the problem of class imbalance, this result may be reflective of the underrepresentation of Manhattan distance in the dataset

Hypothesis Testing

GitHub

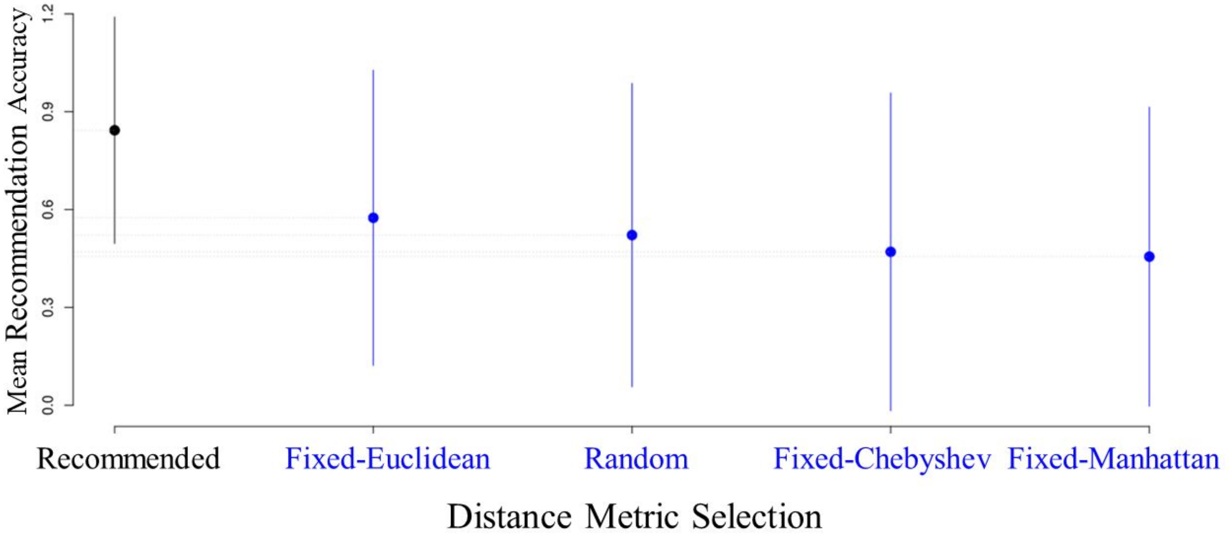
Recommendation Accuracy (RA)

Distance Metric Selection	Mean Recommendation Accuracy
Recommended (Ours)	**83.60%**
Fixed – Chebyshev	47.02%
Fixed – Euclidean	57.46%
Fixed – Manhattan	45.56%
Random	52.15%

- Compares the clustering quality relative to the best- and worst-performing distance metrics (Zhu et al., 2020)

$$RA = \frac{DBS_{\text{rec}} - DBS_{\text{worst}}}{DBS_{\text{best}} - DBS_{\text{worst}}}$$

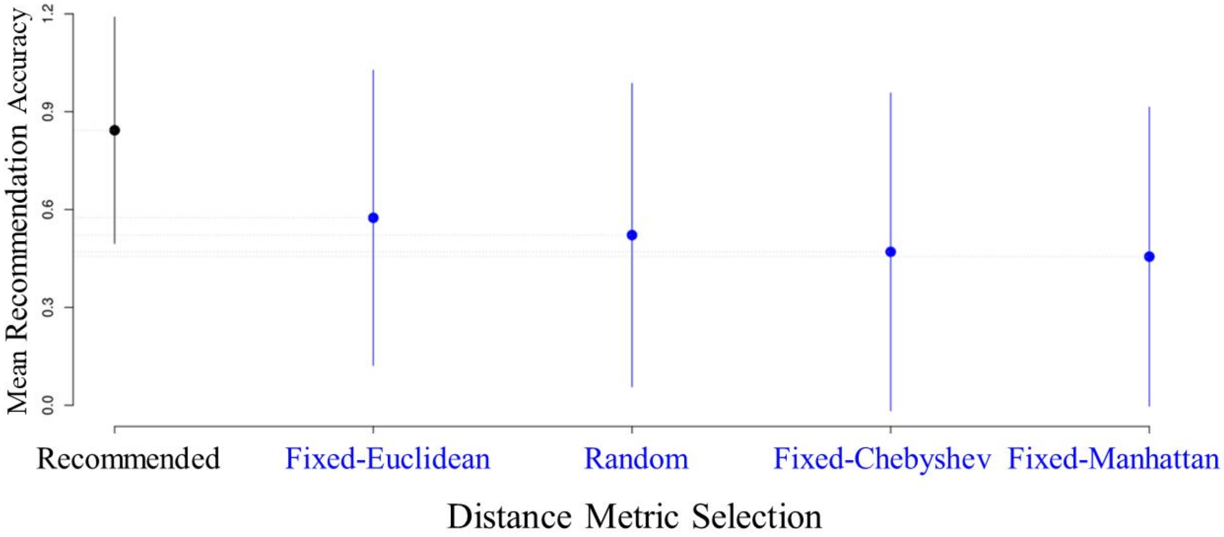
GitHub



Scott-Knott Effect Size Difference Test (Tantithamthavorn et al., 2017)

The mean RA of using our meta-learning model is significantly different from the mean RA of using fixed or random distance metric selection methods

GitHub



Scott-Knott Effect Size Difference Test (Tantithamthavorn et al., 2017)

Our meta-learning model has the lowest standard error of the mean at 0.0344

Euclidean: 0.0448 | Manhattan: 0.0454 | Random: 0.0461 | Chebyshev: 0.0482

GitHub

Conclusion

GitHub

Conclusion

- We explored the use of **(1) general**, **(2) statistical**, **(3) information-theoretic**, **(4) structural**, and **(5) complexity** meta-features in building a **random forest** model that automatically recommends a distance metric for *k*-means clustering

- The model registered an accuracy of **70.59%**

- Limiting the feature set to only the **top 25 most important meta-features** increased the accuracy to **71.57%** (+0.98%)

GitHub

Conclusion

- The fine-grained analysis using SHAP showed that the **mean of the sparsity** registered the **highest feature importance globally**

- While the prediction of the minority class (Manhattan) posed a difficulty despite the application of borderline SMOTE, the **recommendation accuracy of the built meta-learning model is significantly higher compared to using fixed and randomly chosen distance metrics**

GitHub

Distance Metric Recommendation for *k*-Means Clustering: A Meta-Learning Approach

Mark Edward M. Gonzales, Lorene C. Uy, Jacob Adrianne L. Sy & Macario O. Cordel, II
{mark_gonzales, lorene_c_uy, jacob_adrianne_l_sy, macario.cordel}@dlsu.edu.ph
De La Salle University, Manila, Philippines

Project Page

Category	No. of Meta-Features	Meta-Features	
		Abbreviation	Description
General	5	Attr-to-Inst Ratio	Ratio between the number of attributes and instances [26]
		Inst-to-Attr Ratio	Ratio between the number of instances and attributes [27]
		Num Attr	Number of attributes [28]
		Num Binary Attr	Number of binary attributes [28]
		Num Instances	Number of instances [28]
Statistical	32	Canonical Corr †	Canonical correlations of data [29]
		Correlation †	Absolute value of the correlation of distinct dataset column pairs [30]
		Covariance †	Absolute value of the covariance of distinct dataset attribute pairs [30]
		Eigenvalues †	Eigenvalues of covariance matrix from dataset [31]
		IQ Range †	Interquartile range (IQR) of each attribute [32]
		Kurtosis †	Kurtosis of each attribute [28]
		Median Abs Dev †	Median Absolute Deviation (MAD) adjusted by a factor [31]
		Mean †	Mean value of each attribute [33]
		Median †	Median value from each attribute [33]
		Num Correlated Attr	Number of distinct highly correlated pair of attributes [34]
		Num Outliers	Number of attributes with at least one outlier value [35]
		SD †	Standard deviation of each attribute [33]
		Skewness †	Skewness for each attribute [28]
		Sparsity †	(Possibly normalized) sparsity metric‡ for each attribute [34]
		Trimmed Mean †	Trimmed mean of each attribute [33]
		Variance †	Variance of each attribute [30]
Information-Theoretic	2	Concentration Coeff †	Concentration coefficient of each pair of distinct attributes [36]
		Shannon's Entropy †	Shannon's entropy for each predictive attribute [28]
Structural	10	1-Itemset Min, Q1, Q2, Q3, Max	Minimum, first quartile, second quartile, third quartile, and maximum of one itemset meta-feature [37]
		2-Itemset Min, Q1, Q2, Q3, Max	Minimum, first quartile, second quartile, third quartile, and maximum of two itemset meta-feature [37]
Complexity	3	Ave Num Feat per PCA Dim	Average number of features per PCA dimension [38]
		Ave Num PCA Dim per Point	Average number of PCA dimensions per points [38]
		PCA-to-Orig Dim Ratio	Ratio of the PCA dimension to the original dimension [38]

* The meta-feature descriptions are taken from the API documentation [39] of the package PyMFE [40]

† The mean and standard deviation (abbreviated as SD) of these meta-features across all attributes were extracted.

‡ Let n be the number of instances in the dataset and $\phi(a)$ be the number of distinct values under the attribute a.
The sparsity metric is given by $\frac{1}{n-1}\left(\frac{n}{\phi(a)}-1\right)$. It is measure of the degree of discreteness [14].