Supplementary Material of "AutoCluster: Meta-learning Based Ensemble Method for Automated Unsupervised Clustering"

1 The summary of meta-features

Meta-features are crucial for meta-learning that help determine which algorithm or configuration is used for AutoCluster. Therefore, in order to provide a more comprehensive characterization of datasets, the proposed CME extracts 5 new clustering-oriented meta-features from data distribution through Hopkins Statistic and landmarker through different clustering algorithms, which are collected from unlabeled data. Moreover, it also extracts 19 traditional meta-features based on basic structure (simple meta-features), statistical information, and PCA. The summary of meta-features chosen for AutoCluster is listed in Table 1.

Table 1. The implemented meta-features in AutoCluster.

Category	No.	Meta-features
Simple Meta-features	1	number of instances
	2	log number of instances
	3	number of features
	4	log number of features
	5	dataset ratio
	6	log dataset ratio
	7	inverse dataset ratio
	8	inverse log dataset ratio
Statistical Meta-features	9	Kurtosis max
	10	Kurtosis min
	11	Kurtosis mean
	12	Kurtosis std
	13	Skewness max
	14	Skewness min
	15	Skewness mean
	16	Skewness std
PCA Meta-features	17	pca 95%
	18	pca kurtosis first pc
	19	pca skewness first pc
Data Distribution	20	Hopkins Statistic
Landmarker Meta-features	21	the distance to closest cluster center (K-means)
	22	the number of leaves (Agglomerative Clustering)
	23	maximum reachability distance (OPTICS)
	24	maximum core distance (OPTICS)

2 The pseudocode of AutoCluster

AutoCluster is mainly composed of Clustering-oriented Meta-feature Extraction (CME) enhanced meta-learning and Multi-CVIs Clustering Ensemble Construction (MC^{2}EC). For CME, we extract traditional and clustering-oriented meta-features from data distribution and landmarker. The performance data with multiple CVIs is collected for meta-decision data and meta-auxiliary data, which include 3 internal CVIs: Calinski-Harabasz Index (CHI), Davies-Bouldin Index (DBI) and Silhouette Coefficient (SC), and 1 external CVI: Adjusted Rand index 5 (ARI). For MC^{2}EC, it optimizes hyperparameters of promising algorithms suggesting from CME-enhanced meta-learning under SC, CHI, DBI metrics respectively through Bayesian optimization and combines these clustering results to construct ensemble model through Majority Voting. For a new dataset D_{n+1}, as the pseudo code of AutoCluster in online phase shown in Algorithm 1.

Algorithm 1 AutoCluster

Input: new dataset D_{n+1}

Output: optimal data partition π^*

compute meta-features for given dataset D_{n+1}
compute the distance of with datasets in meta-data
select promising algorithms A^{ARI}, A^{SC}, A^{CHI}, A^{DBI} of dataset with min $\{d_i^F\}_{i=1}^n$ for CVIs
incorporate A^{ARI} with A^{SC}, A^{CHI}, A^{DBI} respectively as intersection
set $\pi = \{\}$
for each $cvi \in [SC, CHI, DBI]$ **do**
    $\{\pi_i^{cvi}\}_{i=1}^{n_{cvi}} = GS(A^{cvi})$ # search the promising configuration through grid search
    $\pi = \pi \cup \{\pi_i^{cvi}\}_{i=1}^{n_{cvi}}$
end for
$\pi^* = MV(\pi)$ # combine promising configuration under CVIs through majority voting
return π^*

3 Optimization space in AutoCluster

Table 2. The algorithms and hyperparameter space.

Algorithm Type	Hyperparameter names (values)
K-means	n_clusters: range(2, 30) algorithm: 'auto', 'full', 'elkan' init: 'k-means++', 'random' precompute_distances: 'auto', True, False tol: [1e-5, 1e-2]
Affinity Propagation	damping: [0.5, 1] convergence_iter: range(10, 50) affinity: 'precomputed', 'euclidean'
Mean Shift	cluster_all: True, False min_bin_freq: range(1, 10) bin_seeding: True, False bandwidth: [bandwidth-bandwidth/2, bandwidth+bandwidth/2]
Agglomerative Clustering	n_clusters: range(2, 30) affinity: 'euclidean', 'l1', 'l2', 'manhattan', 'cosine' linkage'ward', 'complete', 'average', 'single' compute_full_tree'auto', True, False
DBSCAN	eps: [0.1, 30] min_samples: range(1, int(X.shape[0] / 30)) algorithm: 'auto', 'ball_tree', 'kd_tree', 'brute' leaf_size: range(1, 60)
Birch	Threshold: [0.1, 2] branching_factor: range(10, 200) n_clusters: range(2, 30)

AutoCluster includes six different clustering algorithms: K-means, Affinity Propagation, Mean Shift, Agglomerative Clustering, DBSCAN, and Birch. They are implemented by scikit-learn. The hyperparameter space are depicted in Table 2. The distance between hyperparameter configuration pair of individuals in MC^{2}EC is based on this hyperparameter space.

4 The visualization of the clustering results on some test datasets of AutoCluster

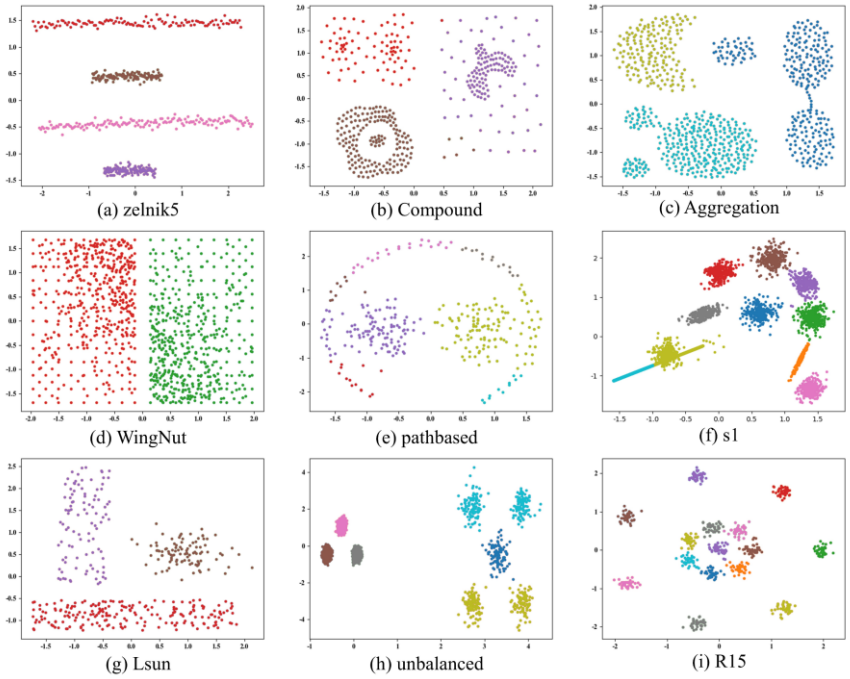

Fig. 1. Clustering result visualization of some test datasets.

5 The running time of AutoCluster

When given new datasets, AutoCluster performs three steps: distance computation for algorithm selection (S1), hyperparameter optimization for high-quality clustering model (S2), and clustering ensemble for automated model construction (S3). Therefore, the runtime experienced by users is bounded by these steps. In order to verify the efficiency of AutoCluster for new datasets, we evaluate the runtime on 33 test datasets as Table 3. From the table, most test datasets can complete automated clustering model construction in low runtime from few seconds to minutes. The smaller datasets (e.g. the number of data points is lower than 1000) generally finish these steps in a minute. For example, No. 10 (*dietary_survey_IBS*) dataset, which has 400 data points and 42 dimensions, only uses 24.57 seconds. Additionally, the larger datasets also can finish in an acceptable runtime. For example, No. 23 (*s3*) dataset, which has 5000 data points and 2 dimensions, uses 763.075 seconds. Therefore, the result shows that AutoCluster runs in an efficient manner for users to automatically build an appropriate clustering model.

Table 3. The runtime of AutoCluster on 33 test datasets.

No.	T1	T2	T3	Data points	Dimensions
1	4.644	41.801	96.096	3000	2
2	1.042	9.769	10.158	788	2
3	1.052	9.804	26.676	804	2
4	1.106	10.594	13.512	800	3
5	0.808	16.267	13.778	625	4
6	0.55	5.947	2.17	399	2
7	1.331	11.07	47.482	1000	2
8	1.308	11.52	4.813	1000	2
9	1.726	21.063	15.431	1232	2
10	0.916	20.308	3.346	400	42
11	2.327	252.86	12.089	1024	32
12	0.313	0.915	0.566	240	2
13	1.324	8.425	3.398	1000	2
14	0.142	24.75	0.343	100	2
15	0.279	18.607	0.379	212	3
16	1.057	9.974	5.737	800	3
17	0.477	23.806	2.798	373	2
18	0.511	21.684	1.367	400	2
19	0.519	341.687	9.149	404	3
20	0.39	1.015	0.761	300	2
21	0.77	14.706	1.518	600	2
22	10.03	111.691	1246.094	5000	2
23	9.241	115.914	568.068	5000	2
24	9.695	112.514	640.866	5000	2
25	1.329	11.681	13.463	1000	2
26	1.032	9.745	13.597	770	2
27	0.516	21.032	1.766	400	3
28	12.786	196.681	5520.524	6500	2
29	1. 365	11.75	48.185	1016	2
30	0.383	14.59	0.63	299	2
31	0.65	16.293	1.547	512	2
32	0.305	37.158	0.881	238	2
33	0.154	19.183	0.456	101	16

6 The correlation of meta-features and algorithm selection

In order to verify the important role of meta-features in AutoCluster, we employ Pearson Correlation Coefficient to measure the linear correlation of single meta-feature and the total number of selected times of 6 algorithms under each CVI. The result is shown in Fig. 2. We can observe that all correlation coefficients are less than 0.5 except No. 5 (*Dataset Ratio*) meta-feature under ARI metric. It reflects the weak correlation between single meta-feature and algorithm selection for each CVI. Therefore, it is infeasible to select promising clustering algorithms using single or few meta-features such as the small number of instances or high dimension, which roughly illustrates the importance of meta-features in AutoCluster.

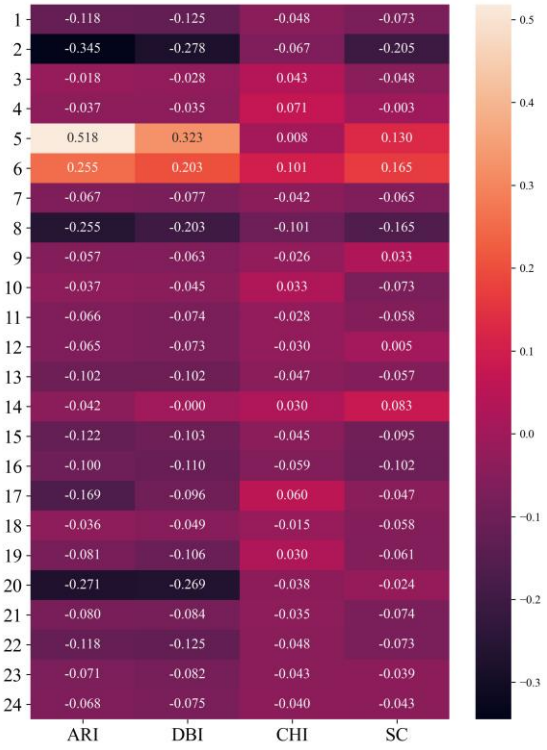

Fig. 2. The correlation of meta-features and algorithm selection under each CVI. The y-axis represents the 24 meta-features and the order is from Table 1.

In this appendix, we also present the rest experiments of the correlation between meta-features and algorithm selection. Fig. 3 and Fig. 4 illustrate the correlation of meta-features and algorithm selection under each CVI. They are SC, ARI, CHI, and DBI respectively. Here, we leverage Pearson Correlation Coefficient (PCC) to measure their correlation. For each CVI, we compute the PCC between meta-features of the datasets selecting any algorithm and not selecting the one respectively. We also observe that the

weak correlation between single meta-feature and clustering algorithm selection since most PCC values are less than 0.5. Therefore, it is important for AutoCluster to consider multiple meta-features to suggest promising algorithms for MC^{2}EC.

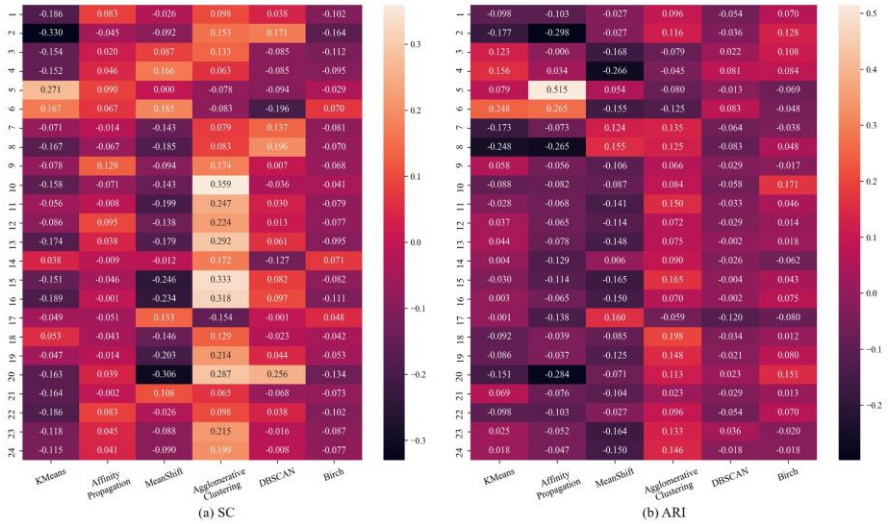

Fig. 3. The correlation of meta-features and algorithm selection under SC and ARI metric.

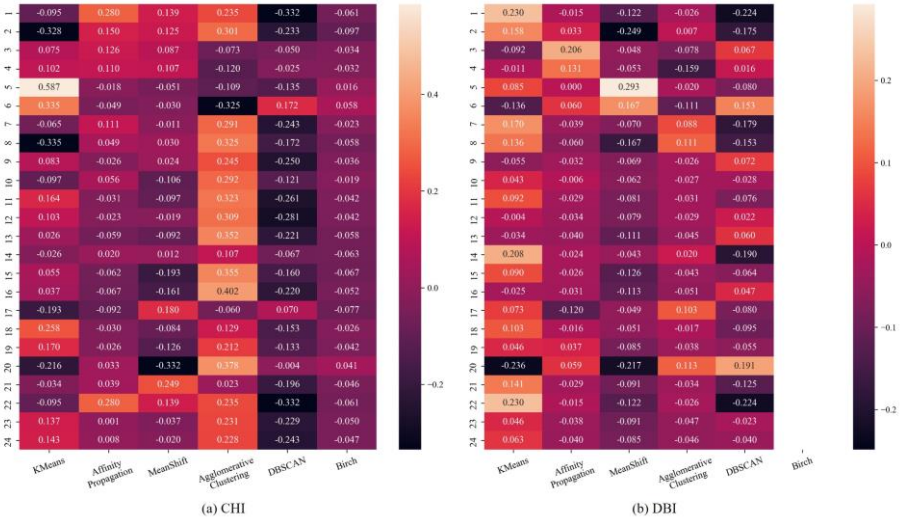

Fig. 4. The correlation of meta-features and algorithm selection under CHI and DBI metric.

Then, we conduct the F-test for other three CVIs with the approach similar with Section 2.3.1. Fig. 5, Fig. 6 and Fig. 7 show the results of SC, ARI, and DBI respectively. We indicate in red the first meta-feature of our proposed CME in the ordering of meta-

features importance. We also observe that CME is important to algorithm selection in AutoCluster under these three CVIs. Note that the dataset selecting Birch algorithm under DBI is so less that we cannot conduct F-test on it, and we dose not present its result.

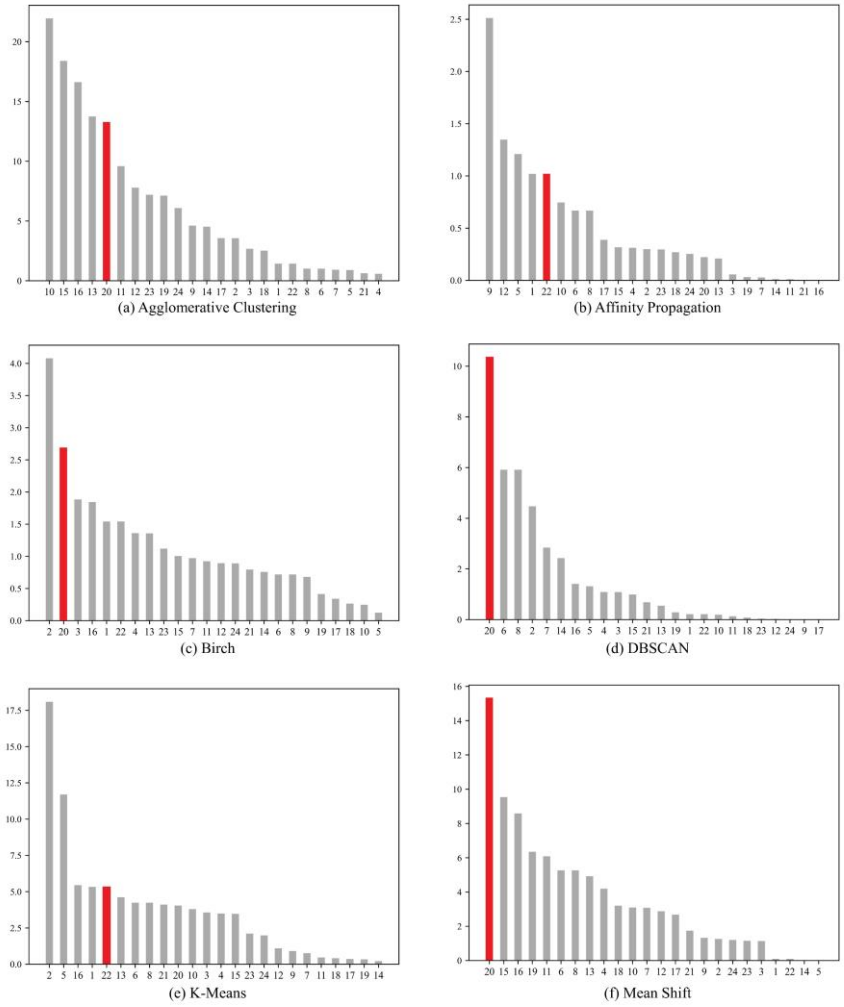

Fig. 5. The F-statics of meta-features grouped by selected algorithm under SC metric.

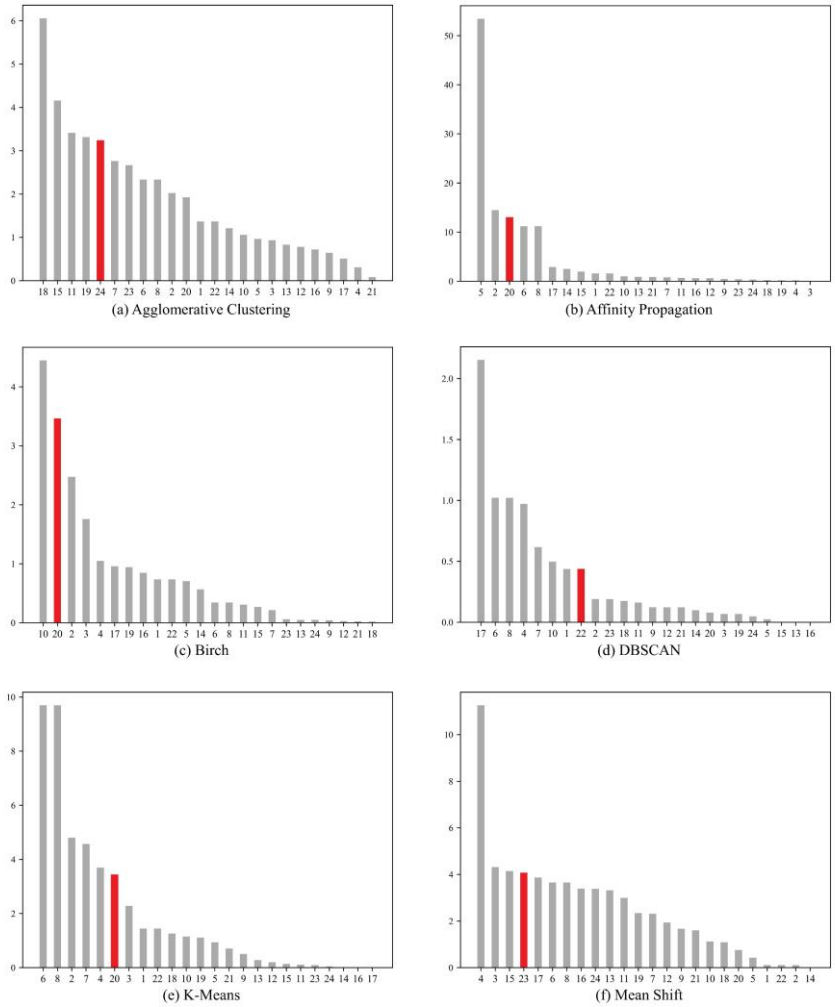

Fig. 6. The F-statics of meta-features grouped by selected algorithm under ARI metric.

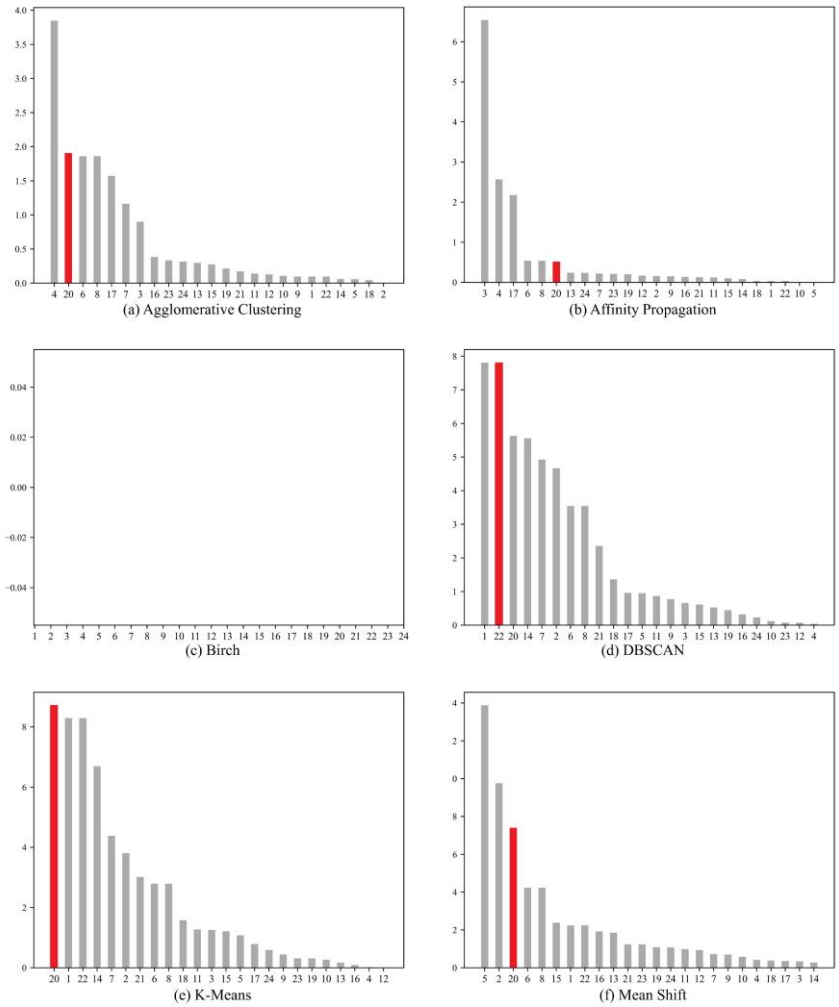

Fig. 7. The F-statics of meta-features grouped by selected algorithm under DBI metric.