Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Gene expression

Single-Cell RNA-seq Data using Evolutionary Multiobjective Ensemble Pruning

Xiangtao Li [1,2], Shixiong Zhang [2] and Ka-Chun Wong [2,*]

[1]Department of Computer Science and Information Technology, Northeast Normal University, Changchun, Jilin, China
[2]Department of Computer Science, City University of Hong Kong, Hong Kong.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Algorithm 1. Pseudocode of evolutionary multiobjective ensemble pruning (EMEP)

Input: input parameters:

1. multiple cluster results $\pi = \{\pi_1, \pi_2, \cdots, \pi_d\}$;
2. Number of subproblems is N (population size);
3. N weight vector $\{\lambda^1, \cdots, \lambda^N\}$;
4. Neighborhood size of each subproblem, denoted as T;

Output: output result:

1. Pareto optimal set $\hat{P}$;

Initialize the population P and transform the population into the binary space $P \in \{0,1\}^d$.
Calculate the neighborhood index B for each individual.
Each population member is randomly assigned with one of the consensus functions from the pool and the associate basic cluster algorithms are chosen randomly from the corresponding pool of values.
Determine those three objective functions f_1, f_2, f_3 for each subproblem in P.
while *a stopping criteria is not satisfied* **do**
  V=Mutation(P);
  U=Crossover(V,P);
  Transform the new population U into the binary space $U \in \{0,1\}^d$.
  Calculate those three objective functions for each subproblem.
  Update of neighboring solutions
  If the target vector is better than the trial vector, then the target vector is randomly reinitialized with a new consensus functions and associated basic cluster algorithms from the respective pools.
Return the Pareto optimal set $\hat{P}$ including all non-dominated individuals with respect to those three objective functions.

Algorithm 2. Pseudocode of Updating Step

Input: input parameters:

1. Number of subproblems is N;
2. The neighborhood index B for each subproblem with the size T;
3. The original population P and the trial vectors U

while *a stopping criteria is not satisfied* **do**
  for $k = 1 \rightarrow N$ **do**
    /*update of neighboring solutions*/
    for $j \in B_k$ **do**
      if $g^{te}(u_k|\lambda^k) < g^{te}(p_j|\lambda^k)$ **then**
        $p_j \leftarrow u_k$;
        $f(p_j) \leftarrow f(u_k)$;
      else
        Adaptive select the consensus functions and basic cluster algorithm from the pool or from the successful combination.

Table S1 The average NMI of parameter analysis on F and CR under the same objective function evaluations.

	CR=0.1	CR=0.2	CR=0.3	CR=0.4	CR=0.5	CR=0.6	CR=0.7	CR=0.8	CR=0.9	CR=1
F=0.1	0.838626	0.831503	0.827302	0.814678	0.830644	0.826168	0.801562	0.829867	0.844924	0.822557
F=0.2	0.830181	0.837879	0.818781	0.839157	0.827056	0.814822	0.826287	0.816301	0.831184	0.834267
F=0.3	0.847664	0.816051	0.837796	0.841339	0.826801	0.830286	0.828055	0.8262	0.831502	0.810505
F=0.4	0.855827	0.835946	0.842831	0.812739	0.838894	0.819818	0.835361	0.81899	0.820459	0.836439
F=0.5	0.833455	0.838683	0.828457	0.808867	0.853008	0.844579	0.848413	0.840127	0.820971	0.846156
F=0.6	0.825096	0.827828	0.8116	0.839515	0.843698	0.83921	0.840223	0.835399	0.818862	0.8546
F=0.7	0.832278	0.82947	0.837608	0.819015	0.846245	0.839458	0.821232	0.829557	0.821863	0.834525
F=0.8	0.831768	0.822987	0.85069	0.842357	0.829677	0.82781	0.835713	0.848118	0.833512	0.84511
F=0.9	0.832278	0.82947	0.837608	0.819015	0.846245	0.839458	0.821232	0.829557	0.821863	0.834525
F=1	0.831768	0.822987	0.85069	0.842357	0.829677	0.82781	0.835713	0.848118	0.833512	0.84511

Table S2 The average ARI of parameter analysis on F and CR under the same objective function evaluations.

	CR=0.1	CR=0.2	CR=0.3	CR=0.4	CR=0.5	CR=0.6	CR=0.7	CR=0.8	CR=0.9	CR=1
F=0.1	0.81457	0.810692	0.80141	0.797295	0.79414	0.797295	0.78786	0.804085	0.832435	0.806137
F=0.2	0.803362	0.807071	0.798515	0.81774	0.815544	0.794995	0.796669	0.798175	0.805904	0.8109
F=0.3	0.815878	0.792087	0.818071	0.822538	0.802998	0.809438	0.803949	0.801049	0.79919	0.810155
F=0.4	0.843436	0.813603	0.827488	0.794695	0.823497	0.801623	0.808584	0.802111	0.793766	0.811417
F=0.5	0.805342	0.814669	0.814615	0.790505	0.837791	0.818642	0.816933	0.815719	0.791435	0.833654
F=0.6	0.797296	0.81088	0.789834	0.80717	0.811727	0.821	0.819	0.81795	0.794527	0.822245
F=0.7	0.808833	0.810048	0.815677	0.790717	0.826933	0.829005	0.801367	0.803093	0.800313	0.814037
F=0.8	0.810433	0.799227	0.82797	0.81412	0.810115	0.805643	0.819602	0.839167	0.807985	0.813392
F=0.9	0.808833	0.810048	0.815677	0.790717	0.826933	0.829005	0.801367	0.803093	0.800313	0.814037
F=1	0.810433	0.799227	0.82797	0.81412	0.810115	0.805643	0.819602	0.839167	0.807985	0.813392

Table S3 The average NMI values versus different values of T in EMEP for six single-cell RNA-seq data. “Avg." denotes the average values of normalized mutual information (NMI)

	T=2	T=4	T=6	T=8	T=10	T=20	T=50	T=100
Buettner	0.860719	0.862945	0.869774	0.81211	0.87003	0.60622	0.620988	0.686052
Deng	0.781781	0.822847	0.781781	0.781781	0.820732	0.781781	0.756725	0.778108
Ginhoux	0.521157	0.550777	0.523998	0.452759	0.473807	0.544972	0.507192	0.470398
Pollen	0.963891	0.967428	0.95708	0.969881	0.965564	0.954194	0.969881	0.954194
Ting	1	1	1	1	0.960394	1	1	1
Treutlin	0.835824	0.930968	0.930968	0.892349	0.895031	0.892349	0.8556	0.844691
Avg	0.827229	0.855827	0.843934	0.818147	0.830926	0.796586	0.785064	0.788907

Table S4 The average ARI values versus different values of T in EMEP for six single-cell RNA-seq data. “Avg." denotes the average values of adjusted rand index (ARI)

	T=2	T=4	T=6	T=8	T=10	T=20	T=50	T=100
Buettner	0.902217	0.901637	0.901308	0.856167	0.901925	0.602612	0.606477	0.739498
Deng	0.550459	0.720188	0.550459	0.550459	0.659619	0.556678	0.486952	0.57881
Ginhoux	0.516764	0.532459	0.51165	0.44419	0.47992	0.535385	0.502803	0.407508
Pollen	0.957555	0.953982	0.951644	0.95952	0.960646	0.952583	0.955365	0.942806
Ting	1	1	1	1	0.963382	1	1	1
Treutlin	0.862251	0.952348	0.952348	0.916927	0.930375	0.916927	0.872448	0.885424
Avg	0.798208	0.843436	0.811235	0.787877	0.815978	0.760697	0.737341	0.759008

Table S5 Performance of different clustering algorithms on the 55 synthetic datasets in terms of normalized mutual information (NMI)

	Knocked-out genes	Noise	LCE	KM	SC	CDP	t-SNE	SCC	ECC	SIMLR	MPSSC	EMEP
1	100	0	1	1	1	1	1	1	1	1	1	1
2	100	0.05	1	1	1	0.829031	1	1	1	1	1	1
3	100	0.1	1	0.80185	1	0.853019	1	1	1	1	1	1
4	100	0.15	0.982283	0.982283	1	0.664977	0.939262	1	1	1	1	1
5	100	0.2	0.821164	0.28927	1	0.359183	0.74605	1	0.931534	0.982283	1	1
6	100	0.25	0.574493	0.144494	0.822876	0.284568	0.445544	0.614319	0.704538	0.639358	0.61296	1
7	100	0.3	0.382863	0.031451	0.876069	0.174239	0.340061	0.789164	0.560253	0.648353	0.781012	0.91517
8	100	0.35	0.153692	0.07866	0.609128	0.046294	0.140645	0.642225	0.277481	0.486384	0.650823	0.773141
9	100	0.4	0.077871	0.033942	0.529521	0.018229	0.101648	0.515014	0.208252	0.208624	0.526731	0.615017
10	100	0.45	0.121884	0.034752	0.452489	0.062976	0.02837	0.411722	0.086226	0.126701	0.405221	0.528553
11	100	0.5	0.095149	0.07041	0.243026	0.013702	0.036614	0.285257	0.05788	0.047658	0.249527	0.220924
12	200	0	1	1	1	1	1	1	1	1	1	1
13	200	0.05	1	1	1	1	1	1	1	1	1	1
14	200	0.1	1	0.855995	1	0.850453	1	1	1	1	1	1
15	200	0.15	1	0.797035	1	1	0.982283	1	1	1	1	1
16	200	0.2	1	0.716633	1	0.951915	0.884911	1	1	1	1	1
17	200	0.25	0.907565	0.653348	1	0.680617	0.796205	1	0.828603	1	1	1
18	200	0.3	0.753655	0.212135	0.982283	0.570484	0.639363	0.982283	0.817817	0.982283	0.982283	1
19	200	0.35	0.510791	0.240692	0.934315	0.327357	0.508777	0.923479	0.684233	0.83028	0.913943	1
20	200	0.4	0.368054	0.109648	0.777846	0.256162	0.435019	0.706944	0.588357	0.706867	0.710934	0.946951
21	200	0.45	0.263373	0.04205	0.725446	0.058524	0.185175	0.661152	0.493808	0.578992	0.656455	0.900645
22	200	0.5	0.138574	0.054155	0.650444	0.102413	0.186625	0.645567	0.300244	0.314175	0.644661	0.780113
23	300	0	1	1	1	1	1	1	1	1	1	1
24	300	0.05	1	1	1	1	1	1	1	1	1	1
25	300	0.1	1	1	1	1	1	1	1	1	1	1
26	300	0.15	1	0.807317	1	1	1	1	1	1	1	1
27	300	0.2	1	1	1	0.744531	0.941113	1	1	1	1	1
28	300	0.25	1	0.460854	1	0.721136	0.799439	1	0.982283	1	1	1
29	300	0.3	0.792838	0.639609	1	0.458816	0.702053	1	0.846232	1	1	1
30	300	0.35	0.694922	0.23418	0.982283	0.564473	0.554123	1	0.73939	1	1	1
31	300	0.4	0.645855	0.177264	0.80185	0.401577	0.556515	1	0.698192	0.964603	0.982283	1
32	300	0.45	0.403729	0.015415	0.916629	0.186877	0.465646	0.89329	0.654862	0.951921	0.89329	1
33	300	0.5	0.322434	0.079138	0.916402	0.218858	0.281644	0.824508	0.410734	0.635014	0.804875	0.951996
34	400	0	1	1	1	1	1	1	1	1	1	1
35	400	0.05	1	1	1	1	1	1	1	1	1	1
36	400	0.1	1	0.80867	1	1	1	1	1	1	1	1
37	400	0.15	1	0.804198	1	1	1	1	1	1	1	1
38	400	0.2	1	1	1	0.766252	0.934298	1	1	1	1	1
39	400	0.25	1	0.735906	0.805087	0.740031	0.87762	1	1	1	1	1
40	400	0.3	0.934298	0.080239	1	0.593148	0.729026	1	0.870976	1	1	1
41	400	0.35	0.757625	0.116043	0.982283	0.419083	0.740531	1	0.765589	1	1	1
42	400	0.4	0.699254	0.258123	0.982283	0.420736	0.561895	1	0.809464	0.982283	1	1
43	400	0.45	0.686228	0.247342	0.982283	0.209695	0.582436	0.982283	0.736177	1	0.982283	1
44	400	0.5	0.488204	0.181314	0.71028	0.31696	0.370578	0.969565	0.652856	0.91662	0.969565	1
45	500	0	1	1	1	1	1	1	1	1	1	1
46	500	0.05	1	0.843319	1	1	1	1	1	1	1	1
47	500	0.1	1	0.802381	1	1	1	1	1	1	1	1
48	500	0.15	1	0.849052	1	1	1	1	1	1	1	1
49	500	0.2	1	1	1	0.840296	0.939262	1	1	1	1	1
50	500	0.25	1	0.762781	0.801784	0.7208	0.80437	1	1	1	1	1
51	500	0.3	0.756917	0.084663	1	0.660897	0.727776	1	0.931534	1	1	1
52	500	0.35	0.749172	0.3928	0.80185	0.622064	0.610329	1	0.759871	1	1	1
53	500	0.4	0.797501	0.247887	0.982283	0.614866	0.661044	1	0.770659	1	1	1
54	500	0.45	0.686325	0.2337	0.982283	0.487164	0.609501	0.982283	0.768683	1	0.982283	1
55	500	0.5	0.665111	0.196696	0.982283	0.459129	0.533315	1	0.713756	1	1	1
Avg.			0.767851	0.531049	0.913333	0.623119	0.715983	0.924165	0.811827	0.890953	0.922711	0.956955

Table S6 Performance of different clustering algorithms on the 55 synthetic datasets in terms of adjusted rand index (*ARI*.

	Knocked-out genes	Noise	LCE	KM	SC	CDP	t-SNE	SCC	ECC	SIMLR	MPSSC	EMEP
1	100	0	1	1	1	1	1	1	1	1	1	1
2	100	0.05	1	1	1	0.77411	1	1	1	1	1	1
3	100	0.1	1	0.625378	1	0.768661	1	1	1	1	1	1
4	100	0.15	0.986599	0.986599	1	0.526595	0.947518	1	1	1	1	1
5	100	0.2	0.839374	0.203185	1	0.272252	0.732552	1	0.935249	0.986599	1	1
6	100	0.25	0.484596	0.091841	0.847611	0.244227	0.403767	0.59167	0.671976	0.580059	0.593675	1
7	100	0.3	0.256146	0.010977	0.897286	0.135908	0.344769	0.813332	0.529284	0.641422	0.802949	0.922739
8	100	0.35	0.072026	0.051293	0.559906	0.020752	0.10299	0.613693	0.263999	0.464522	0.626346	0.73879
9	100	0.4	0.028316	0.004868	0.484151	-0.00076	0.071514	0.48122	0.151268	0.20203	0.502102	0.605363
10	100	0.45	0.050115	0.018963	0.391996	0.016724	0.01171	0.35918	0.06666	0.099067	0.36785	0.527004
11	100	0.5	0.000503	0.045342	0.210333	-0.00343	0.019009	0.245393	0.031185	0.035656	0.233429	0.209578
12	200	0	1	1	1	1	1	1	1	1	1	1
13	200	0.05	1	1	1	1	1	1	1	1	1	1
14	200	0.1	1	0.704666	1	0.762478	1	1	1	1	1	1
15	200	0.15	1	0.693957	1	1	0.986599	1	1	1	1	1
16	200	0.2	1	0.612333	1	0.960354	0.886895	1	1	1	1	1
17	200	0.25	0.899809	0.578536	1	0.579494	0.765848	1	0.811143	1	1	1
18	200	0.3	0.755792	0.154596	0.986599	0.512517	0.586038	0.986599	0.819252	0.986599	0.986599	1
19	200	0.35	0.469133	0.151702	0.947343	0.223271	0.467737	0.934506	0.663444	0.850398	0.922219	1
20	200	0.4	0.316556	0.05927	0.769089	0.19943	0.416771	0.702208	0.515173	0.664043	0.711927	0.960216
21	200	0.45	0.20693	0.006348	0.709911	0.021346	0.15878	0.629659	0.454557	0.540687	0.628003	0.911065
22	200	0.5	0.011312	0.032551	0.633084	0.071119	0.159908	0.623863	0.275263	0.307711	0.63118	0.776657
23	300	0	1	1	1	1	1	1	1	1	1	1
24	300	0.05	1	1	1	1	1	1	1	1	1	1
25	300	0.1	1	1	1	1	1	1	1	1	1	1
26	300	0.15	1	0.63674	1	1	1	1	1	1	1	1
27	300	0.2	1	1	1	0.655424	0.947657	1	1	1	1	1
28	300	0.25	1	0.328127	1	0.653212	0.78949	1	0.986599	1	1	1
29	300	0.3	0.756182	0.569101	1	0.330672	0.666463	1	0.832799	1	1	1
30	300	0.35	0.629912	0.193147	0.986599	0.520434	0.510932	1	0.698118	1	1	1
31	300	0.4	0.577963	0.125596	0.625378	0.345872	0.523584	1	0.646785	0.973333	0.986599	1
32	300	0.45	0.33618	-0.00137	0.934086	0.028353	0.443996	0.908937	0.621228	0.960344	0.908937	1
33	300	0.5	0.315527	0.039263	0.933293	0.155644	0.279344	0.849463	0.385192	0.630706	0.827177	0.960609
34	400	0	1	1	1	1	1	1	1	1	1	1
35	400	0.05	1	1	1	1	1	1	1	1	1	1
36	400	0.1	1	0.639423	1	1	1	1	1	1	1	1
37	400	0.15	1	0.630363	1	1	1	1	1	1	1	1
38	400	0.2	1	1	1	0.681999	0.947379	1	1	1	1	1
39	400	0.25	1	0.634498	0.632209	0.658924	0.875707	1	1	1	1	1
40	400	0.3	0.947379	0.016646	1	0.518797	0.684767	1	0.855424	1	1	1
41	400	0.35	0.741454	0.055285	0.986599	0.318633	0.703648	1	0.707239	1	1	1
42	400	0.4	0.617176	0.237896	0.986599	0.231321	0.50807	1	0.798955	0.986599	1	1
43	400	0.45	0.62787	0.224871	0.986599	0.052025	0.540886	0.986599	0.687483	1	0.986599	1
44	400	0.5	0.474029	0.168019	0.566348	0.248841	0.352498	0.973474	0.621509	0.934104	0.973474	1
45	500	0	1	1	1	1	1	1	1	1	1	1
46	500	0.05	1	0.692342	1	1	1	1	1	1	1	1
47	500	0.1	1	0.626519	1	1	1	1	1	1	1	1
48	500	0.15	1	0.698387	1	1	1	1	1	1	1	1
49	500	0.2	1	1	1	0.736491	0.947518	1	1	1	1	1
50	500	0.25	1	0.672725	0.625235	0.676343	0.789737	1	1	1	1	1
51	500	0.3	0.726769	0.043124	1	0.57809	0.688417	1	0.935249	1	1	1
52	500	0.35	0.661401	0.278421	0.625378	0.535428	0.529102	1	0.721422	1	1	1
53	500	0.4	0.728793	0.185533	0.986599	0.537868	0.63519	1	0.744012	1	1	1
54	500	0.45	0.639978	0.193081	0.986599	0.40515	0.539642	0.986599	0.749725	1	0.986599	1
55	500	0.5	0.59643	0.160201	0.986599	0.427196	0.500859	1	0.675412	1	1	1
Avg.			0.740986	0.474188	0.896099	0.570578	0.699405	0.921571	0.797375	0.888071	0.921376	0.956582

Table S7 Performance of our proposed algorithm EMEP with different clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP), t-Distributed Stochastic Neighbor Embedding (t-SNE) , Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on six small-scale single-cell RNA-seq datasets by NMI. For each algorithm, a Wilcoxon's signed-rank test is conducted to verify whether the experiment results of best algorithms are better than other algorithms. If p-value is less than 0.05, it denotes that the best algorithm of interest performs significantly better than other algorithms.

	LCE	KM	SC	CDP	t-SNE	SSC	ECC	SIMLR	MPSSC	EMEP
Buettner	0.481831	0.429695	0.802566	0.613695	0.37924	0.763803	0.460942	0.83808	0.834247	**0.862945**
Deng	0.721629	0.745784	0.678172	0.364413	0.705009	0.657895	0.742988	0.751462	0.755436	**0.822847**
Ginhoux	0.431339	0.375828	0.583962	0.023231	0.396178	0.582787	0.430115	0.38777	**0.663596**	0.550777
Pollen	0.950483	0.810659	0.894751	0.423855	0.93429	0.944631	0.906688	0.948059	0.936047	**0.967428**
Ting	0.879946	0.512785	0.862202	0.536255	0.80313	0.975537	0.810118	0.97542	0.975537	**1**
Treutlin	0.79982	0.779984	0.722564	0.384705	0.73455	0.702194	0.60608	0.688089	0.801038	**0.930968**
$+/-/\approx$	6/0/0	6/0/0	5/1/0	6/0/0	6/0/0	5/1/0	6/0/0	6/0/0	5/1/0	N/A
Avg.	0.710841	0.609123	0.757369	0.391026	0.658733	0.771141	0.659488	0.764813	0.82765	**0.855827**

Table S8 Performance of our proposed algorithm EMEP with different clustering algorithms including including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP), t-Distributed Stochastic Neighbor Embedding (t-SNE) , Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on six small-scale single-cell RNA-seq datasets by ARI. For each algorithm, a Wilcoxon's signed-rank test is conducted to verify whether the experiment results of best algorithms are better than other algorithms. If p-value is less than 0.05, it denotes that the best algorithm of interest performs significantly better than other algorithms.

	LCE	KM	SC	CDP	t-SNE	SCC	ECC	SIMLR	MPSSC	EMEP
Buettner	0.317873	0.368972	0.816863	0.497696	0.300073	0.763083	0.447217	0.858609	0.844273	**0.90164**
Deng	0.500416	0.480668	0.500536	0.08703	0.517017	0.380399	0.58065	0.477191	0.478277	**0.72019**
Ginhoux	0.39297	0.281424	0.60951	0.004956	0.407522	0.593723	0.384869	0.323924	**0.73166**	0.532459
Pollen	0.942945	0.558086	0.804438	0.074606	0.874029	0.92916	0.887371	0.940825	0.932761	**0.95398**
Ting	0.776858	0.344062	0.797449	0.407265	0.709367	0.978416	0.67829	0.980272	0.978416	**1**
Treutlin	0.835293	0.831695	0.54818	0.162507	0.54869	0.524231	0.453429	0.511448	0.611716	**0.95235**
$+/-/\approx$	6/0/0	6/0/0	5/1/0	6/0/0	6/0/0	5/1/0	6/0/0	6/0/0	5/1/0	N/A
Avg.	0.627726	0.477484	0.679496	0.205677	0.55945	0.694835	0.571971	0.682045	0.76285	**0.84344**

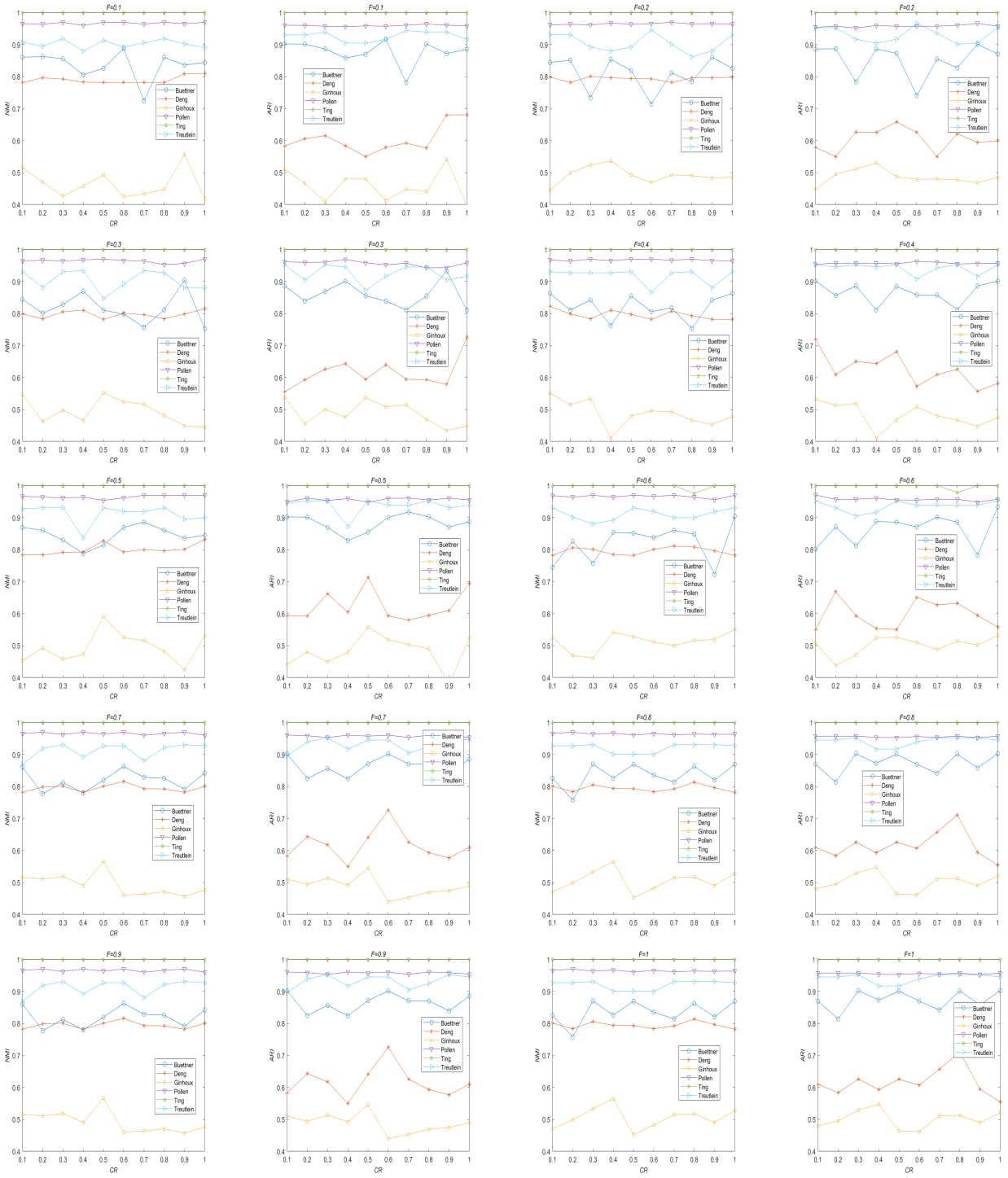


Figure S1 The performance of various F values from 0.1 to 1 with CR values from 0.1 to 1 on six single-cell RNA-seq datasets including Buettner, Deng, Ginhoux, Pollen, Ting, and Treutlein. The performance is measured using normalized mutual information (*NMI*) and adjusted rand index (*ARI*).

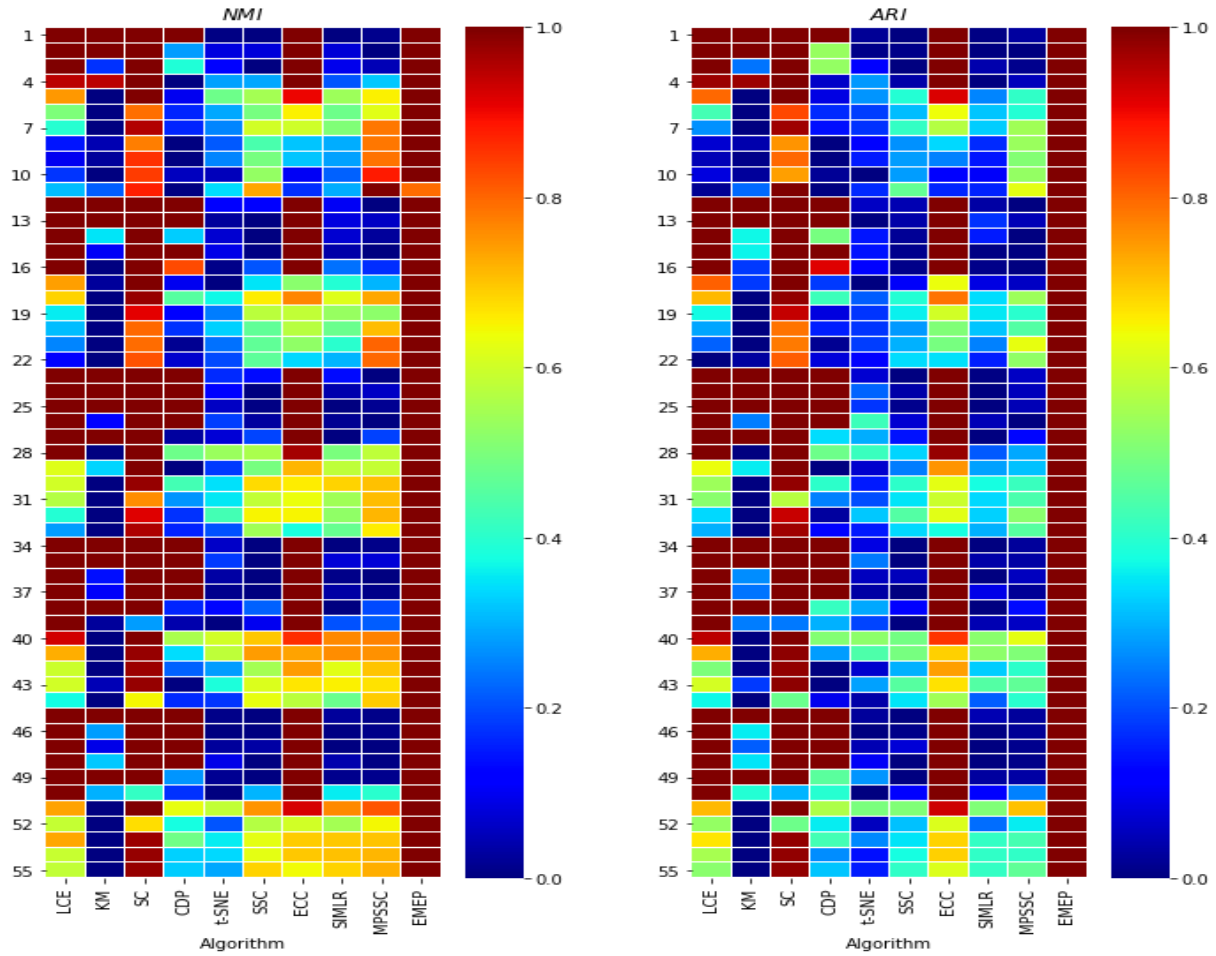


Figure S2 The performance of EMEP and other nine clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP) , t-Distributed Stochastic Neighbor Embedding (t-SNE), Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on 55 simulated datasets.The performance is measured using the normalized mutual information (*NMI*) and adjusted rand index (*ARI*).

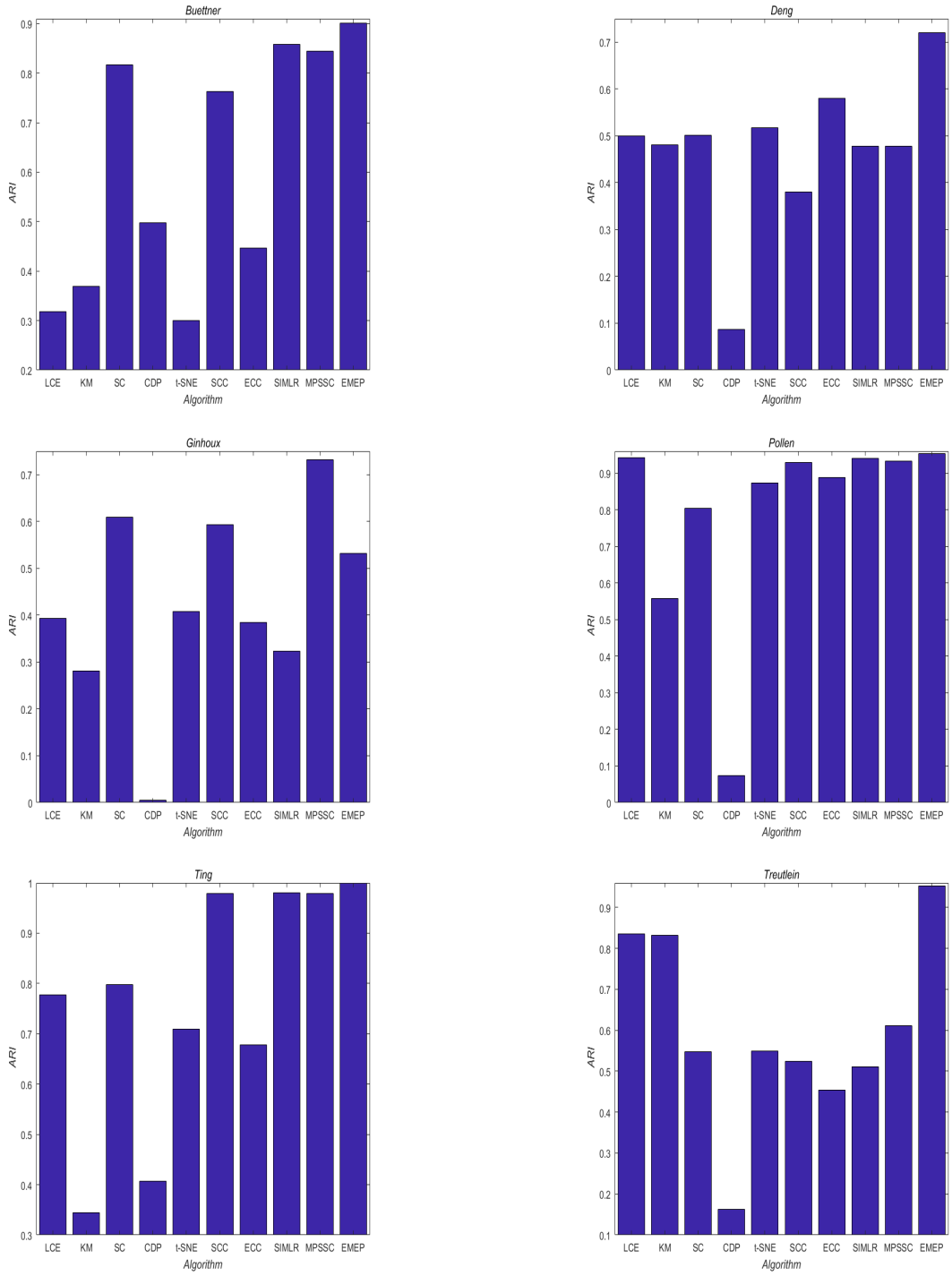


Figure S3 The performance of EMEP and other nine clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP) , t-Distributed Stochastic Neighbor Embedding (t-SNE), Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on the six small-scale single-cell RNA-seq datasets.The performance is measured using the adjusted rand index (*ARI*).

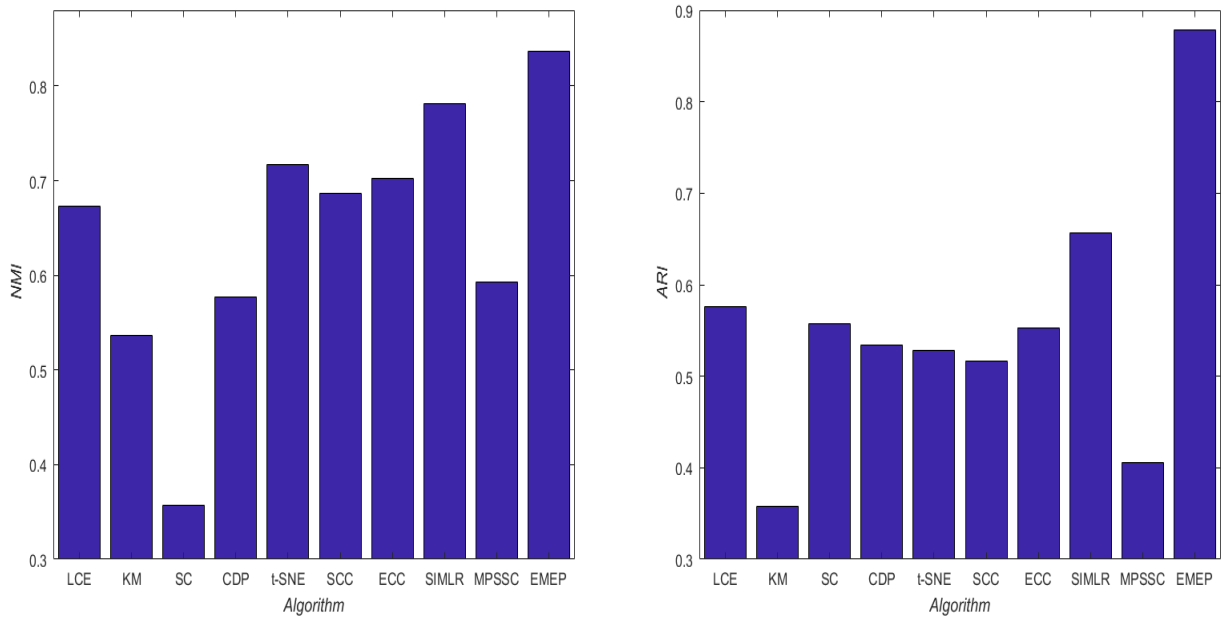
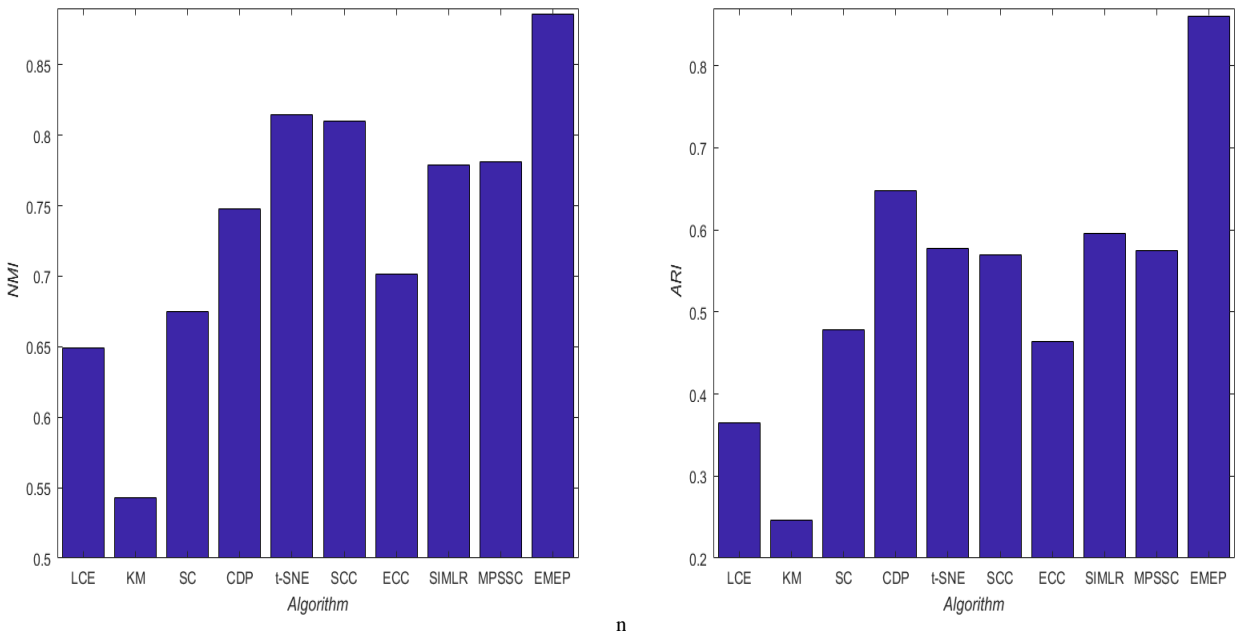


Figure S4 The performance of EMEP and other nine clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP) , t-Distributed Stochastic Neighbor Embedding (t-SNE), Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on one large-scale single-cell RNA-seq datasets.The performance is measured using normalized mutual information (*NMI*) and adjusted rand index (*ARI*).



n

Figure S5 The performance of EMEP and other nine clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP) , t-Distributed Stochastic Neighbor Embedding (t-SNE), Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on Islet single cells from the NCBI Gene Expression Omnibus (GEO) repository. The performance is measured using the normalized mutual information (*NMI*) and adjusted rand index (*ARI*).

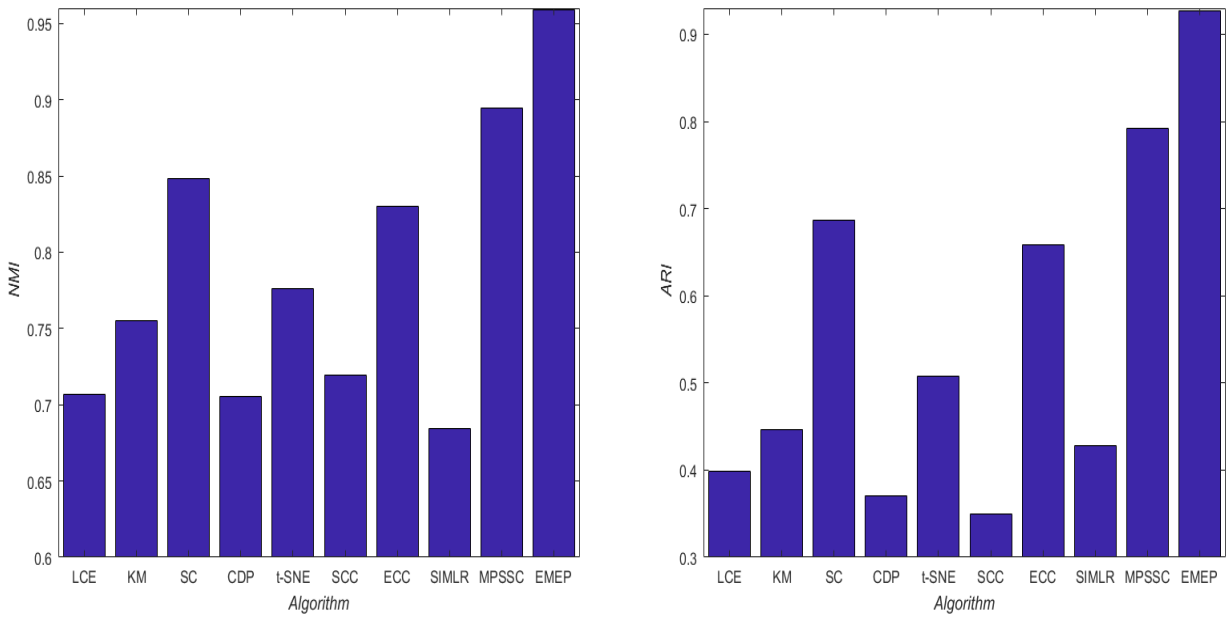


Figure S6 The performance of EMEP and other nine clustering algorithms including Link-based Cluster Ensemble (LCE), Entropy-based Consensus Clustering (ECC), Spectral Clustering (SC), K-means Clustering (KM), clustering by fast search and find of density peaks (CDP) , t-Distributed Stochastic Neighbor Embedding (t-SNE), Single-Cell Interpretation via Multikernel Learning (SIMLR), Sparse Spectral Clustering (SSC), and Spectral clustering based on learning similarity matrix (MPSSC) on human cancer cells dataset.The performance is measured using the Normalized Mutual Information (*NMI*) and adjusted rand index (*ARI*).

Effect of different combinations of objective functions

DB index is a function of the sum ratio of within-cluster scatter to between-cluster separation. It can be computed as follows:

$$f_4 = \frac{1}{K}\sum_{i=1}^{K} max_{i\neq j}(\frac{\delta_i + \delta_j}{d(c_i, c_j)}) \quad (1)$$

where $d(c_i, c_j)$ is the Euclidean distance between centroid of clusters c_i and c_j and $\delta_i = \sqrt{\frac{1}{n_k}\sum_{p\in C_i}\sum_{q=1}^{K} \mid x_{pq} - c_{iq} \mid^2}$ is dispersion measure of a cluster C_i. The DB index value must be minimized to achieve proper clustering. In other words, small value represents a better clustering.

The nex objective function is Dunn index, which determines the ratio between the minimal intercluster distance to maximal intracluster distance. It is applied to find compact and well-separated clusters. It can be described as:

$$f_5 = \min_i\{\min_j(\frac{\min_{x\in C_i, y\in C_j} d(x,y)}{\max_k\{\max_{x,y\in C_k} d(x,y)\}})\} \quad (2)$$

where $d(x, y)$ is the Euclidean distance between two samples x and y in different clusters C_i and C_j, respectively [S1].

To show the effectiveness of those three objective function as the final objective functions, we compare our proposed algorithm with those 31 different combinations for objective functions. Since individual objective function f_3 is to minimize the number of chosen basic partition clusters for regularization, which cannot optimize the clustering as the sole objective function. Therefore, we remove the individual objective function f_3 from the 31 different combinations. All these objectives are optimized using evolutionary multiobjective ensemble pruning algorithm (EMEP). In this experiment, our proposed algorithm is compared with those 29 different combinations of objective functions based on six real-world single-cell RNA-seq datasets to evaluate its complementary strengths and nonredundancy. The experiment results are summarized in Supplementary Table S9-S10 for the external measurements including normalized mutual information (*NMI*) and adjusted rand index (*ARI*). In Supplementary Table S9-S10, "D" denotes the overall deviation of partitioning f_1; "C" denotes the compactness of clustering f_2; "NC" denotes the number of chosen basic partition clusters f_3; "DB" denotes the DB index f_4; "$Dunn$" denotes the Dunn index f_5. Figure S7 provides the visualization of those results.

For individual objective function, we can found that $D + C + NC$ can provide better solutions than those individual objective functions, especially for DB. From Supplementary Table S9-S10, we can also find that D and C perform better than DB and $Dunn$, which demonstrate that the objective functions D and C is more suitable for address single-cell RNA-seq over DB and $Dunn$. Therefore, in this study, we choose the objective functions D and C into the objective functions.

To show the nonredundancy of our proposed combination of objective function, we can firstly compare $D + C + NC$ with individual single objective D and C. From Supplementary Table S9-S10, it is pointed out that the performance of $D + C + NC$ is superior to individual single objective. Then, we also compare $D + C + NC$ with the pair of objective functions $D + C$, $D + NC$, and $C + NC$. $D + C + NC$ also obtain the better solutions than the pair of objective functions. Based on the analysis, we can conclude that our proposed combination of objective function $D + C + NC$ don't exist the redundant objective function.

For the ensemble pruning, the objectives involve both maximizing the generalization performance and minimizing the number of clusters for regularization. Unfortunately, those two objectives are usually conflicting; the optimal decision needs to be enabled as the trade-off between those two objectives. In this case, it would be ideal to regard ensemble pruning as a multi-objective problem rather than a single-objective problem. For the first goal, we consider two objective functions including the overall cluster deviation and The within-cluster compactness. The overall cluster deviation represents the total sum of distances between data points and their corresponding cluster centers. The within-cluster compactness measures the average distance between every pairs of data points in the same cluster. Those two objective functions are just precisely that a well-separated cluster needs two characteristics, which is also the complementary strengths of our proposed algorithm. For the second goal of ensemble pruning, the third objective function is designed to minimize the number of chosen basic partition clusters, that except to obtain a good cluster using small number of basic partition clusters. Moreover, from the Supplementary Table S9-S10, we can conclude that $D + C + NC$ perform better than other combinations for objective functions, which also demonstrate the strengths of our proposed combination of objective function $D + C + NC$.

[S1] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures." *In Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pp. 911-916, 2010.

The Time complexity

In this section, the time complexity of proposed algorithm EMEP is analyzed, which include the time complexity of unsupervised dimensionality reduction, and evolutionary multiobjective ensemble pruning.

At the beginning of EMEP algorithm, we first apply the non-negative matrix factorization to project data from the original high dimensional spaces to lower dimensional subspaces as the unsupervised dimensionality reduction. For the non-negative matrix factorization, the main goal is to minimize the objective function by iteratively updating the variables. From the Section 2.2, three different operators including addition operation, multiplication operation, and division operations are used in the algorithm. For the addition operation, the time complexity is $O(2 \times m \times n \times r + 2(m + n) \times r^2)$. The multiplication operation is larger than the addition operation which is used to update the basis matrix W and coefficient matrix H. The time complexity of multiplication operation costs $O(2 \times m \times n \times r + 2(m + n) \times r^2 + (m + n) \times r)$. For the division operation, it is smaller than the addition operation and multiplication operation, which costs $O((m + n) \times r)$. Therefore, after t iterations, the overall cost for NMF is $O(n \times m \times r \times t)$ where n is the number of data, m is the dimension of the features in the original data space, and r is the feature dimension of dataset after factorization.

Then, evolutionary multiobjective ensemble pruning removes some of the multiple cluster results and further improves the generalization performance for clustering. Since the algorithm randomly selects two solutions for mutation and crossover operators and generates the reference point, costing $O(E)$, E is the number of objective function. Then, the offspring individual is obtained to choose some of the multiple cluster results. After that, adaptive selection method is apply to select a consensus function from the pool and then find the corresponding clustering algorithm to produce the final cluster result. Because the pool has three consensus functions, we will randomly choose one consensus function and clustering algorithm. For consensus function, they have the similarity computational complexity, which cost $O(n^2 \times m)$. For the clustering algorithms, three clustering algorithms including *K-means* clustering algorithm (KM), spectral clustering (SC), and clustering by fast search and find of density peaks (CDP) are considered. For the KM algorithm, if the number of clusters k, the problem can be accurately solved in time $O(k \times m \times n \times I)$ with a fixed number I of iterations. For the CDP, it is used to find the cluster center, costing $O(n^2)$. For spectral clustering, it makes use of the eigenvalues of the similarity matrix of the data for clustering,

Table S9 Performance of different subsets on six real-world single-cell RNA-seq datasets by *NMI*. "D" denotes the overall deviation of partitioning f_1; "C" denotes the compactness of clustering f_2; "NC" denotes the number of chosen basic partition clusters f_3; "DB" denotes the DB index f_4; "$Dunn$" denotes the Dunn index f_5.

	D	C	DB	Dunn	D+C	D+NC	C+NC	D+DB	C+DB	Dunn+D
Buettner	0.8515	0.8515	0.1575	0.5183	0.8515	0.8444	0.5390	0.8121	0.7868	0.8515
Deng	0.7924	0.7818	0.3252	0.8240	0.7965	0.7837	0.8062	0.7800	0.7818	0.8009
Ginhoux	0.4305	0.4260	0.4146	0.2614	0.4305	0.4334	0.3729	0.3924	0.4299	0.4159
Pollen	0.9656	0.9699	0.6838	0.9496	0.9585	0.9533	0.9636	0.9639	0.9604	0.9656
Ting	0.9755	1.0000	0.7901	0.8473	0.9784	0.9349	1.0000	1.0000	1.0000	0.9193
Treutlin	0.9310	0.7800	0.6349	0.7680	0.9523	0.8942	0.8286	0.9310	0.8556	0.9074
Avg.	0.8244	0.8015	0.5010	0.6948	0.8280	0.8073	0.7517	0.8132	0.8024	0.8101
	Dunn+C	Dunn+DB	Dunn+NC	DB+NC	D+C+DB	Dunn+D+C	D+C+NC	Dunn+D+DB	D+DB+NC	Dunn+D+NC
Buettner	0.8446	0.4437	0.4918	0.1633	0.8101	0.8356	0.8629	0.8260	0.8549	0.8445
Deng	0.7818	0.7818	0.6881	0.7532	0.7924	0.7754	0.8228	0.7955	0.8043	0.7845
Ginhoux	0.4142	0.4420	0.1023	0.4064	0.5199	0.4417	0.5508	0.4569	0.5135	0.4419
Pollen	0.9617	0.9388	0.9475	0.8839	0.9604	0.9639	0.9674	0.9699	0.9557	0.9658
Ting	0.9478	0.8875	0.7753	0.9106	1.0000	0.9193	1.0000	1.0000	1.0000	1.0000
Treutlin	0.9015	0.8354	0.8406	0.7935	0.8669	0.8443	0.9310	0.9189	0.8804	0.9310
Avg.	0.8086	0.7215	0.6410	0.6518	0.8249	0.7967	0.8558	0.8279	0.8348	0.8280
	Dunn+C+DB	C+DB+NC	Dunn+C+NC	Dunn+DB+NC	D+C+DB+Dunn	D+C+DB+NC	D+C+Dunn+NC	D+DB+Dunn+NC	C+DB+Dunn+Nu	D+C+DB+Dunn+NC
Buettner	0.7349	0.7960	0.5577	0.8698	0.8605	0.6465	0.8080	0.8859	0.8012	0.8700
Deng	0.7818	0.7818	0.8121	0.7650	0.7970	0.7965	0.7818	0.8106	0.7818	0.7837
Ginhoux	0.4202	0.5290	0.4154	0.4612	0.4774	0.5045	0.4217	0.4348	0.5505	0.5189
Pollen	0.9641	0.9589	0.9630	0.9518	0.9560	0.9634	0.9656	0.9639	0.9652	0.9605
Ting	0.9755	1.0000	1.0000	0.7145	1.0000	1.0000	1.0000	0.9755	1.0000	1.0000
Treutlin	0.8358	0.8028	0.7899	0.6009	0.8923	0.8617	0.9310	0.8806	0.8211	0.9074
Avg.	0.7854	0.8114	0.7564	0.7272	0.8305	0.7954	0.8180	0.8252	0.8200	0.8401

Table S10 Performance of different subsets on six real-world single-cell RNA-seq datasets by *ARI*. "D" denotes the overall deviation of partitioning f_1; "C" denotes the compactness of clustering f_2; "NC" denotes the number of chosen basic partition clusters f_3; "DB" denotes the DB index f_4; "$Dunn$" denotes the Dunn index f_5.

	D	C	DB	Dunn	D+C	D+NC	C+NC	D+DB	C+DB	Dunn+D
Buettner	0.8862	0.8862	0.0606	0.3539	0.8862	0.8857	0.5211	0.8562	0.8404	0.8862
Deng	0.5939	0.5788	0.1079	0.6687	0.5946	0.5931	0.7183	0.5788	0.5594	0.6100
Ginhoux	0.4186	0.4090	0.4200	0.1648	0.4186	0.3972	0.3548	0.3962	0.4295	0.3705
Pollen	0.9585	0.9606	0.3117	0.9417	0.9656	0.9526	0.9611	0.9576	0.9413	0.9563
Ting	0.9784	1.0000	0.7106	0.7362	0.9755	0.9184	1.0000	1.0000	1.0000	0.9184
Treutlin	0.9523	0.7000	0.6258	0.7036	0.9310	0.9332	0.8635	0.9523	0.8724	0.9304
Avg.	0.7980	0.7558	0.3727	0.5948	0.7953	0.7800	0.7365	0.7902	0.7738	0.7786
	Dunn+C	Dunn+DB	Dunn+NC	DB+NC	D+C+DB	Dunn+D+C	D+C+NC	Dunn+D+DB	D+DB+NC	Dunn+D+NC
Buettner	0.8859	0.3184	0.4318	0.0248	0.8558	0.8707	0.9016	0.8704	0.8863	0.8856
Deng	0.5505	0.5505	0.4751	0.5254	0.6090	0.5576	0.7202	0.5937	0.6615	0.5939
Ginhoux	0.3657	0.4477	0.0035	0.3314	0.5106	0.4167	0.5325	0.4344	0.4964	0.4450
Pollen	0.9538	0.9320	0.9324	0.8445	0.9428	0.9576	0.9540	0.9588	0.9428	0.9511
Ting	0.9461	0.7891	0.6355	0.8964	1.0000	0.9184	1.0000	1.0000	1.0000	1.0000
Treutlin	0.9103	0.8414	0.9064	0.8259	0.8836	0.8938	0.9523	0.9394	0.8359	0.9523
Avg.	0.7687	0.6465	0.5641	0.5747	0.8003	0.7691	0.8434	0.7995	0.8038	0.8047
	Dunn+C+DB	C+DB+NC	Dunn+C+NC	Dunn+DB+NC	D+C+DB+Dunn	D+C+DB+NC	D+C+Dunn+NC	D+DB+Dunn+NC	C+DB+Dunn+Nu	D+C+DB+Dunn+NC
Buettner	0.7667	0.8400	0.4278	0.9013	0.9017	0.5136	0.8262	0.9174	0.8557	0.9019
Deng	0.5505	0.5505	0.5670	0.5697	0.6094	0.5946	0.5596	0.6433	0.5505	0.5931
Ginhoux	0.4266	0.5194	0.4415	0.4614	0.4680	0.5003	0.4055	0.4269	0.5213	0.5089
Pollen	0.9621	0.9608	0.9457	0.9400	0.9419	0.9592	0.9576	0.9576	0.9515	0.9571
Ting	0.9787	1.0000	1.0000	0.6293	1.0000	1.0000	1.0000	0.9784	1.0000	1.0000
Treutlin	0.8420	0.8459	0.7524	0.4882	0.9169	0.9046	0.9523	0.8959	0.8699	0.9304
Avg.	0.7544	0.7861	0.6891	0.6650	0.8063	0.7454	0.7835	0.8032	0.7915	0.8152

which costs $O(n^3)$. Among those three clustering algorithm, the spectral clustering algorithm has the highest complexity. Therefore, the time complexity $O(n^3)$ is chosen as the worst-case complexity. After that, for all subproblems, the time complexity is $O(N \times (n^2 \times m + n^3))$. Then, to use the neighborhood information to update the population, it needs $O(E \times T)$ underlying operations since its major costs lie in the E calculations for T solutions, where T is the number of the neighborhood for each subproblem. In summary, the computational complexity for the neighborhood updating is $O(E \times N \times T)$. Therefore, the overall time complexity of EMEP is $O(n \times m \times r \times t + N \times (n^2 \times m + n^3 + E \times T))$.

Comparison of EMEP with other NMF variants

We compare our algorithm with the NMF with KL-divergence (NMF_KL) (Lee and Seung, 2001), NMF with α-divergence (NMF$_\alpha$)(Cichocki et al., 2006, 2008), Poisson-based NMF algorithm (PNMF) (Neher et al., 2009), Orthogonal NMF (ONMF) (Stražar et al. 2016), Robust NMF (RNMF) (Guan et al., 2012), NMF based on symmetric information divergence (SINMF) (Devarajan et al., 2015). In this experiment, we employ those NMF variants to replace the Frobenius norm based NMF in EMEP to conduct a fair comparison. The experimental results are summarized in

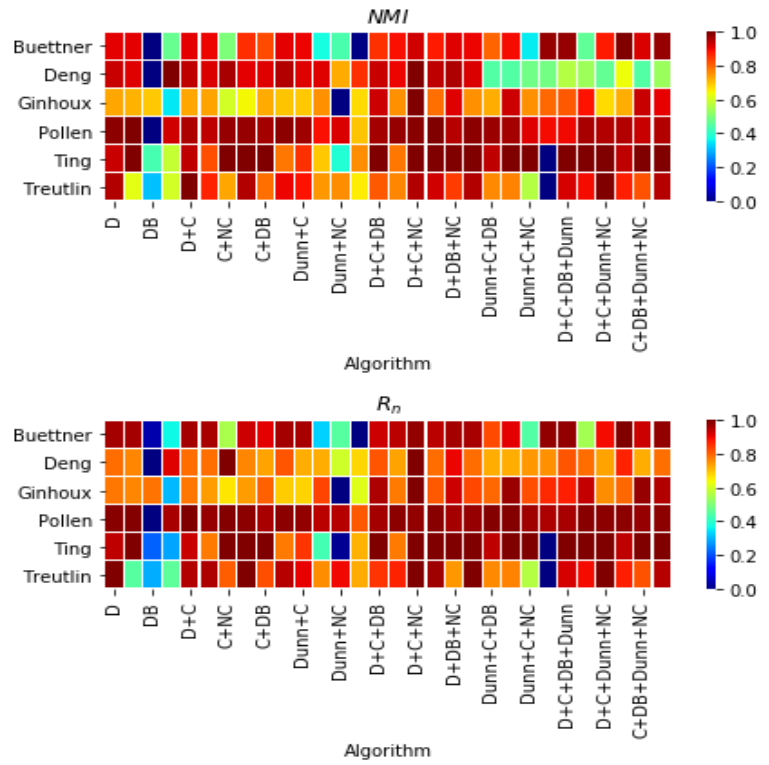


Figure S7 The performance of all different combinations of objective functions on six real-world single-cell RNA-seq datasets on NMI and ARI. The horizontal axis denotes different single objective while the vertical axis denotes six real-world single-cell RNA-seq datasets.

Figure S8 and Supplementary Table S11-S12. For the evaluation metrics *NMI*, EMEP is the best-performing algorithm. Meanwhile, the results of Wilcoxon's signed-rank test in the last two rows indicate that EMEP is significantly better than NMF_KL, NMF_α, PNMF, ONMF, RNMF, and SINMF on 5, 6, 5, 5, 6, and 6 single-cell RNA-seq datasets respectively. It is significantly worse than PNMF and ONMF on one single-cell RNA-seq dataset respectively. NMF_KL, RNMF, SINMF, and NMF_α cannot provide significantly better results than EMEP. Figure S8 depicts the performance of different algorithms measured by NMI. Figure S8(a) visualizes the performance of various NMF methods based on the Normalized Mutual Information (NMI). Figure S8(b) shows the average results of NMI for different NMF methods which demonstrate the robustness of our proposed algorithm.

For adjusted rand index (*ARI*), the experimental results are summarized in Supplementary Table S12 which tabulates the mean values of 30 runs for each method. As observed from Supplementary Table S12, we can conclude several observations: (1) our proposed algorithm EMEP can perform better than six different NMF algorithms on those single-cell RNA-seq datasets while RNMF shows the worst performance. (2) NMF_KL can give better solutions than other traditional algorithms while EMEP is inferior to, equal to, and superior to NMF_KL on 3, 2, and 1 single-cell RNA-seq datasets, respectively. (3) NMF_α, PNMF, RNMF, and SINMF cannot generate any single-cell RNA-seq comparable to EMEP while PNMF and ONMF only can give one better solution than MOCDP. (4) Figure S8 shows the performance of different algorithms measured by the adjusted rand index (*ARI*). Figure S8(c) demonstrates the performance of various NMF methods based on the adjusted rand index (*ARI*). Figure S8(d) shows the average results of *ARI*, similar to *NMI*.

The above observations motivated us to use the Frobenius norm based NMF.

Lee, D. D. and Seung, H. S.(2001). Algorithms for non-negative matrix factorization. *In Advances in neural information processing systems*, 556-562.

Cichocki, A., Amari, S. I., Zdunek, R., Kompass, R., Hori, G., He, Z. (2006, June). Extended SMART algorithms for non-negative matrix factorization. *In International Conference on Artificial Intelligence and Soft Computing* (pp. 548-562). Springer, Berlin, Heidelberg.

Cichocki, A., Lee, H., Kim, Y. D., Choi, S. (2008). Non-negative matrix factorization with α-divergence. *Pattern Recognition Letters*, 29(9), 1433-1440.

Neher, R. A., Mitkovski, M., Kirchhoff, F., Neher, E., Theis, F. J., Zeug, A. (2009). Blind source separation techniques for the decomposition of multiply labeled fluorescence images. *Biophysical journal*, 96(9), 3791-3800.

Guan, N., Tao, D., Luo, Z., Yuan, B. (2012). Online nonnegative matrix factorization with robust stochastic approximation. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7), 1087-1099.

Stražar, M., Žitnik, M., Zupan, B., Ule, J., Curk, T. (2016). Orthogonal matrix factorization enables integrative analysis of multiple RNA binding proteins. *Bioinformatics*, 32(10), 1527-1535.

Devarajan, K., Ebrahimi, N., Soofi, E. (2015, November). A hybrid algorithm for non-negative matrix factorization based on symmetric information divergence. *In Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on* (pp. 1658-1664). IEEE.

Table S11 The performance of EMEP with other methods on six single-cell RNA-seq data by NMI. For each algorithm, a Wilcoxon's signed-rank test is conducted to verify whether the experiment results of best algorithms are better than other algorithms. If p-value is less than 0.05, it denotes that the best algorithm of interest performs significantly better than other algorithms.

	NMF_KL	NMF_α	PNMF	ONMF	RNMF	SINMF	EMEP
Butter	0.782388	0.6946	0.870001	0.869774	0.446184	0.817384	0.862945
Deng	0.781781	0.792552	0.788388	0.798639	0.764535	0.787245	0.822847
Ginhoux	0.554433	0.348142	0.421714	0.441279	0.458475	0.426196	0.550777
Pollen	0.965564	0.961726	0.960643	0.963891	0.866568	0.955959	0.967428
Ting	1	0.975537	0.937989	0.959305	0.921927	0.959305	1
Treutlin	0.927429	0.892928	0.880637	0.90944	0.746077	0.835776	0.930968
Avg.	0.835266	0.777581	0.809895	0.823721	0.700628	0.796977	0.855827
$+/-/\approx$	$5/0/1$	$6/0/0$	$5/1/0$	$5/1/0$	$6/0/0$	$6/0/0$	N/A

Table S12 The performance of EMEP with other methods on six single-cell RNA-seq data by ARI. For each algorithm, a Wilcoxon's signed-rank test is conducted to verify whether the experiment results of best algorithms are better than other algorithms. If p-value is less than 0.05, it denotes that the best algorithm of interest performs significantly better than other algorithms.

	NMF_KL	NMF_α	PNMF	ONMF	RNMF	SINMF	EMEP
Buettner	0.823165	0.7216	0.901783	0.901308	0.398639	0.855143	0.901637
Deng	0.550459	0.621799	0.609009	0.643235	0.632543	0.551468	0.720188
Ginhoux	0.53946	0.359477	0.330322	0.420367	0.493871	0.326661	0.532459
Pollen	0.957555	0.943321	0.943897	0.957555	0.825818	0.94257	0.953982
Ting	1	0.978416	0.946561	0.957433	0.916643	0.957433	1
Treutlin	0.945462	0.903045	0.90492	0.928057	0.803367	0.894639	0.952348
Avg.	0.802683	0.75461	0.772749	0.801326	0.67848	0.754652	0.843436
$+/-/\approx$	$3/2/1$	$6/0/0$	$5/1/0$	$5/1/0$	$6/0/0$	$6/0/0$	N/A

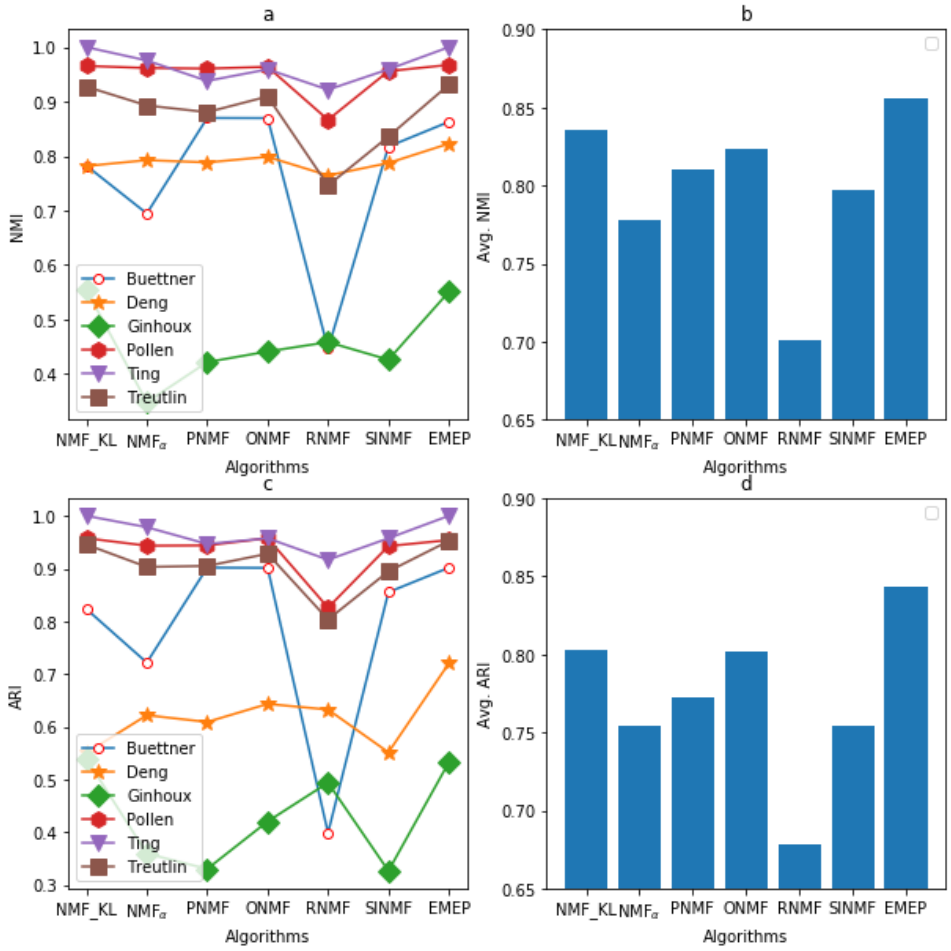
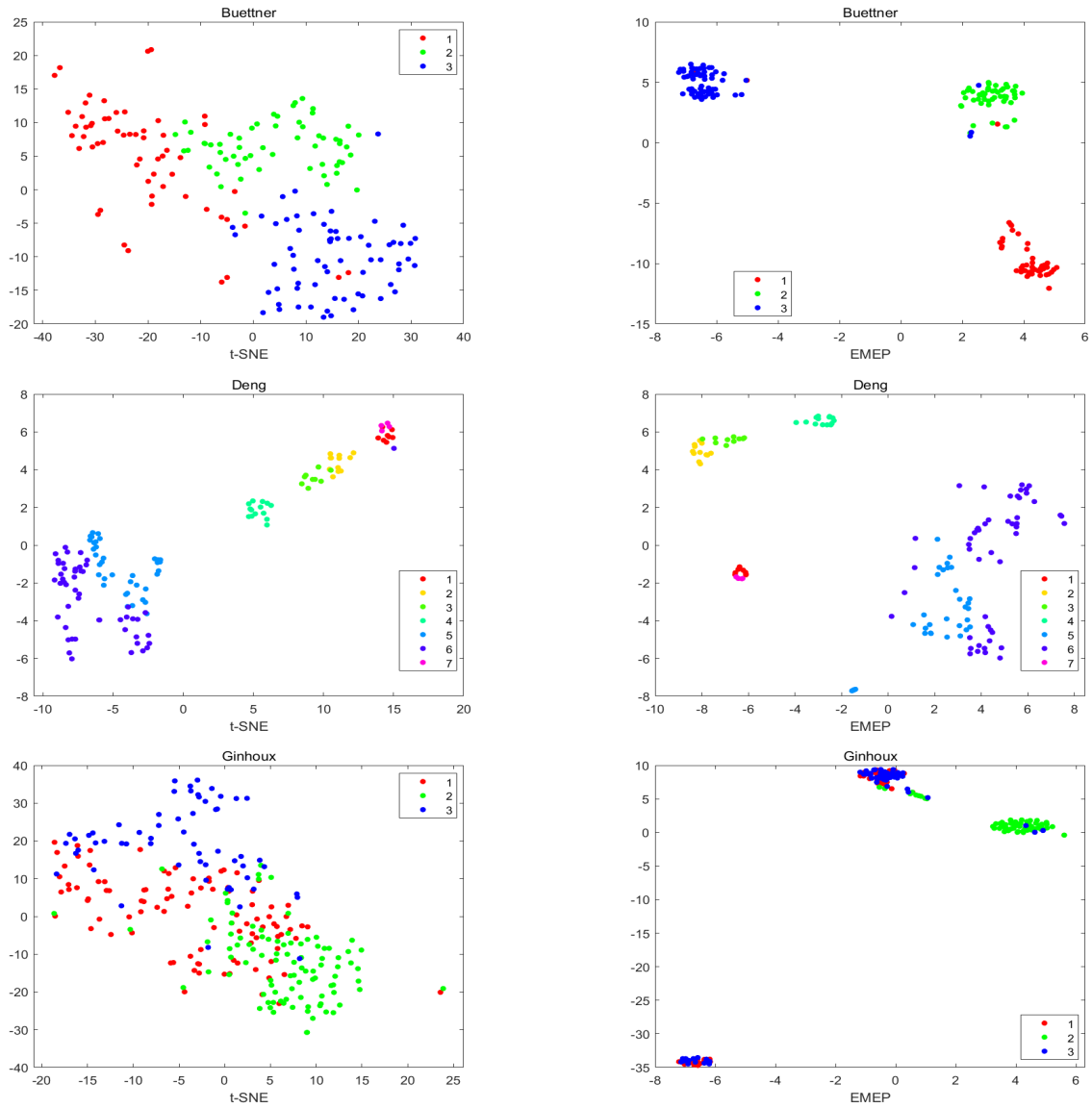
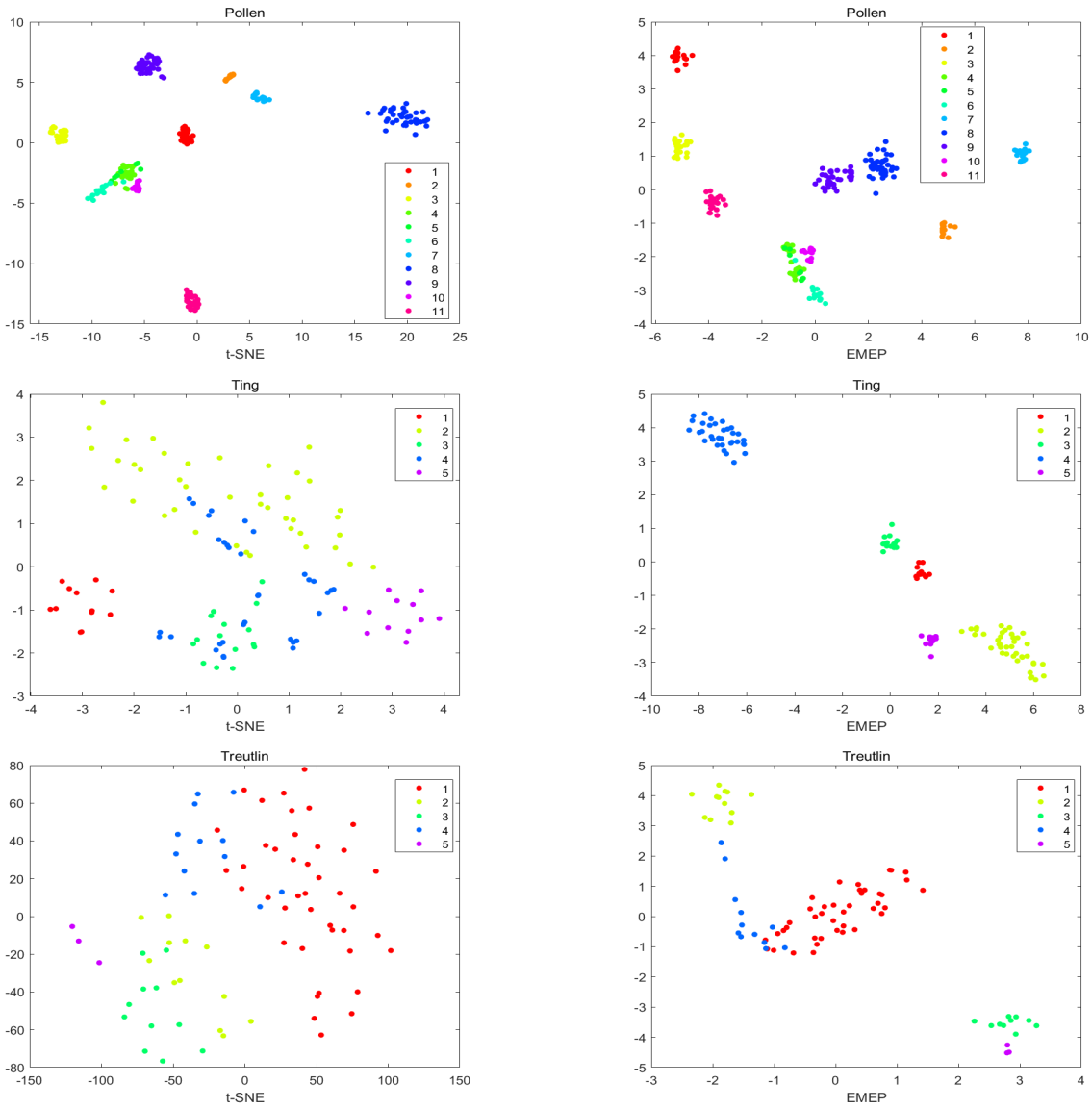


Figure S8 The performance of EMEP with other methods on six single-cell RNA-seq data. (a) The performance is measured by the Normalized Mutual Information (*NMI*); (b) The average value of NMI for different variants of NMF; (c) The performance is measured by the adjusted rand index (*ARI*); (d) The average value of ARI for different variants of NMF.



Figures S9 Comparison of 2D visualization. The axes are in arbitrary units. Each point represents a cell and smaller distances between two cells represent greater similarity. None of the those two methods used the true labels as inputs and the true label information was added in the form of distinct colors to validate the results for Buettner, Deng, and Ginhoux datasets.



Figures S10 Comparison of 2D visualization. The axes are in arbitrary units. Each point represents a cell and smaller distances between two cells represent greater similarity. None of the those two methods used the true labels as inputs and the true label information was added in the form of distinct colors to validate the results for Pollen, Ting, and Treutlin datasets.