iMass: an approximate adaptive clustering algorithm for dynamic data using probability based dissimilarity

Panthadeep BHATTACHARJEE, Pinaki MITRA

Frontiers of Computer Science, DOI: 10.1007/s11704-019-9116-y

Current Problem and original ideas

Problem: Inability of MBSCAN clustering algorithm to handle dynamic updates to dataset.

Existing ideas: Based on our observation no adaptive or incremental extension to MBSCAN exists .

Limitation of MBSCAN:
Building blocks like *iForest*, Mass-matrix involve high computational cost.

Results of MBSCAN on various datasets.

Dataset	#iTrees t	δ_{core}	μ	*iForest* built time (sec)	Mass-matrix built time(sec)	Core/Non-core, Clusters, outliers time (sec)	MBSCAN time(excl. *iForest*)(sec)
Libras	20	5	0.31694	15	16.751	0.004702	16.7695
Segment	20	10	0.52816	25	1130.08	0.136521	1130.26
Wine	20	5	0.370225	16	2.08052	0.001545	2.0915
Seeds	21	7	0.44195	16	3.03881	0.004456	3.05231
Aggregation	24	9	0.426301	17	145.055	0.033157	145.118
Iris	20	7	0.405	18	1.361	0.002541	1.36982
S1	20	9	0.344	18	244.346	0.080631	244.476
S2	20	10	0.23113	16	951.056	0.107407	952.131

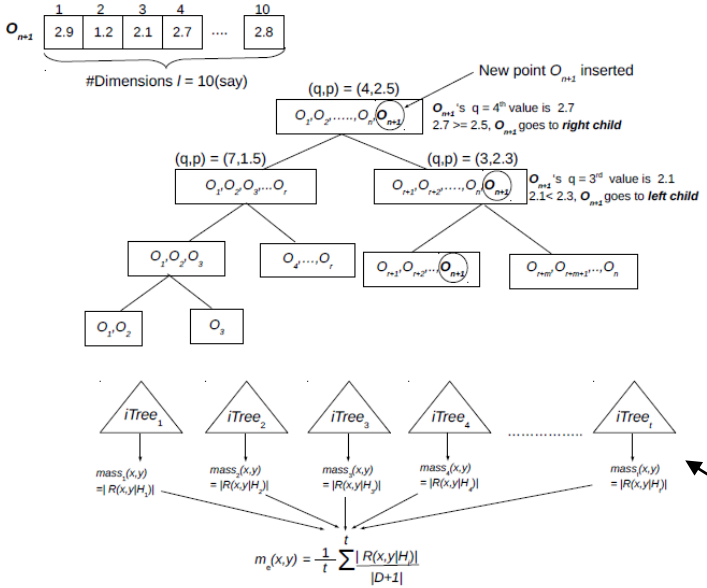
Propose *iMass* clustering algorithm which facilitates point-based insertion adaptively.

Target expensive components: *iForest*, Mass-matrix incrementally to make *iMass* more efficient than MBSCAN.

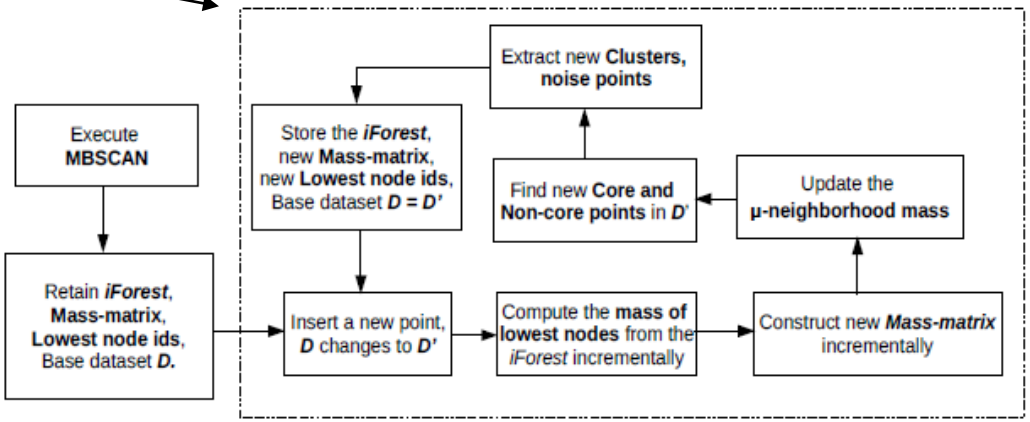
Mathematically deduce a relation to compute the dissimilarity score between any existing pair of points

Percentage of total time required to construct the mass-matrix

Dataset	Libras	Segment	Wine	Seeds	Aggregation	Iris	S1	S2
Percentage of total time for constructing the mass-matrix	99.88	99.98	99.47	99.55	99.95	99.42	99.94	99.88



Compute the node mass incrementally upon new point insertion.



iMass clustering algorithm

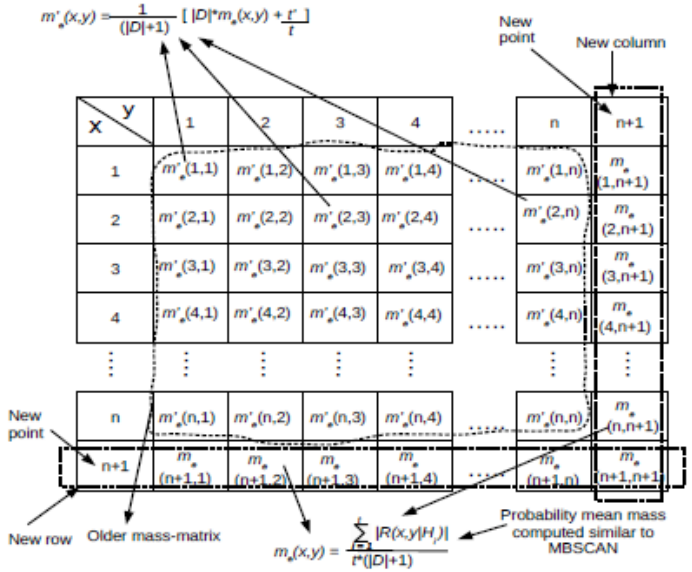
Sequence of execution for the *iMass* algorithm.

iForest construction dynamically

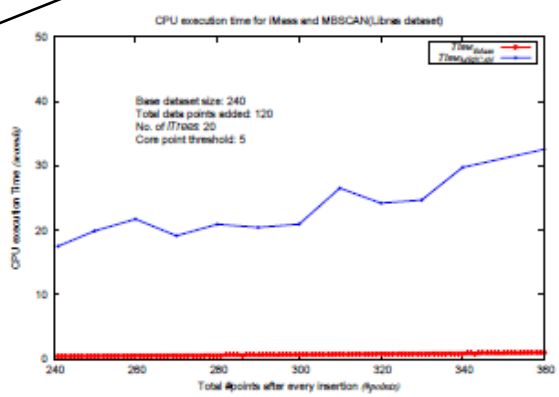
Main Results/Conclusions

Experimental results

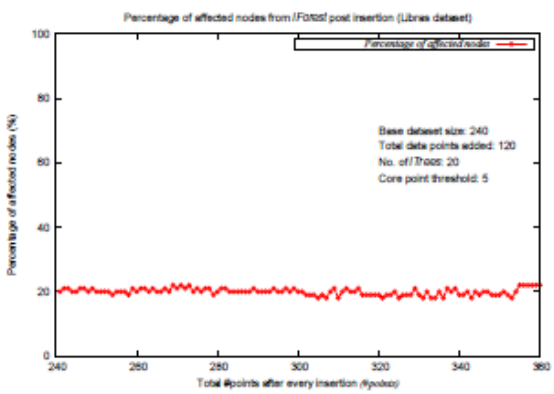
Updated Mass-matrix



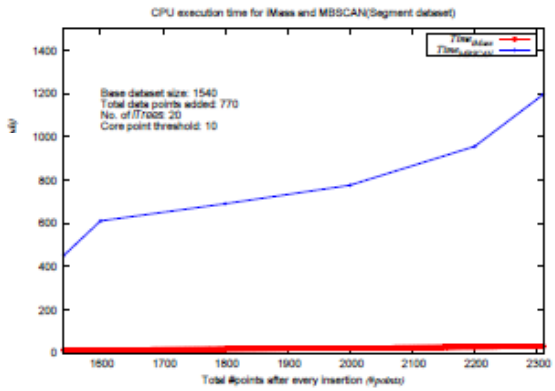
Updated mass-matrix post insertion of a new point.



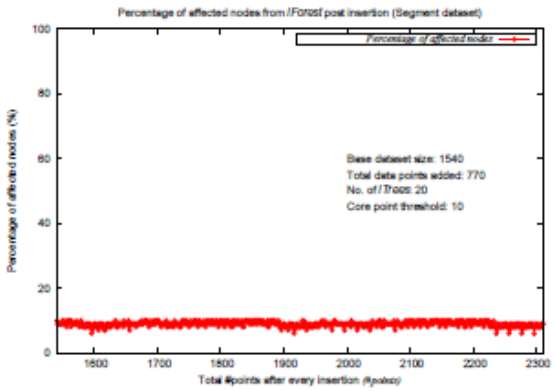
(a) Execution time for *iMass* and MBSCAN after every insertion for Libras dataset



(b) Percentage of affected nodes in the *iForest* after every insertion for Libras dataset.



(c) Execution time for *iMass* and MBSCAN after every insertion for Segment dataset



(d) Percentage of affected nodes in the *iForest* after every insertion for Segment dataset.

Extent of reduction achieved for building *iForest* and mass-matrix incrementally due to *iMass*.

Dataset	D'	*iForest*			Mass-matrix		
		MBSCAN (sec)	*iMass* (sec)	Reduction %	MBSCAN (sec)	*iMass* (sec)	Reduction %
Libras	241	13	0.00258	99.98	4.4741	0.4009	91.03
	270	14	0.002525	99.98	5.1014	0.4953	90.28
	300	14	0.002359	99.98	6.8980	0.6208	90.99
	320	14	0.00317	99.97	10.1717	0.7097	93.02
	360	16	0.001829	99.98	16.5059	0.9130	94.46
Segment	1541	7	0.00196	**99.97**	444.12	13.6027	**96.93**
	1600	7	0.00065	**99.99**	604.434	14.5633	**97.59**
	2000	7	0.00067	**99.99**	769.443	22.516	**97.07**
	2200	7	0.00068	**99.99**	948.945	27.3928	**97.11**
	2310	7	0.00068	**99.99**	1189.24	30.1611	**97.46**
Wine	119	14	0.0015	99.98	0.5356	0.0985	81.60
	125	14	0.00207	99.98	0.6408	0.1079	83.15
	155	13	0.00162	99.98	1.3750	0.1701	86.96
	170	14	0.00150	99.98	1.6561	0.2021	87.79
	178	14	0.002	99.98	2.057	0.2134	89.62
Seeds	141	14	0.00134	**99.99**	0.8938	0.1379	**84.56**
	160	15	0.00181	**99.98**	1.3327	0.1758	**86.80**
	180	16	0.00170	**99.98**	2.1317	0.2246	**89.46**
	200	15	0.00180	**99.98**	2.604	0.2893	**88.88**
	210	16	0.0016	**99.98**	3.1947	0.3515	**97.46**

Efficiency achieved in *iForest* and Mass-matrix construction

Conclusions:
iMass more efficient than MBSCAN.
Cluster quality is approximately similar.