Split and merge tweak in Cross-Entropy Clustering

Krzysztof Hajto

Jagiellonian University

10.02.2017

Table of Contents

Table of Contents

Cross entropy

Given random variables X and Y, the cross entropy is the mean code length of Y using coding optimized for coding X.

$$h^{\times}(Y\|X) := \sum_i g_i \cdot (-\log_2 f_i).$$

Cross entropy

Given random variables X and Y, the cross entropy is the mean code length of Y using coding optimized for coding X.

$$h^{\times}(Y\|X) := \sum_i g_i \cdot (-\log_2 f_i).$$

It extends to continuous variables as

$$H^{\times}(Y\|X) := \int g(y) \cdot (-\ln f(y))dy.$$

Cross entropy

Given random variables X and Y, the cross entropy is the mean code length of Y using coding optimized for coding X.

$$h^{\times}(Y\|X) := \sum_i g_i \cdot (-\log_2 f_i).$$

It extends to continuous variables as

$$H^{\times}(Y\|X) := \int g(y) \cdot (-\ln f(y))dy.$$

By the *cross-entropy of data set X with respect to density f* is given by

$$H^{\times}(X\|f) = -\frac{1}{|X|}\sum_{\mathrm{x}\in X} \ln(f(\mathrm{x})).$$

Cross entropy clustering

Let us consider a splitting of $X \subset \mathbb{R}^N$ into $X_1, \ldots, X_k$. We consider the elements of each cluster X_i to be "coded" by some optimal density f_i out of a family $\mathcal{F}_i$.

Cross entropy clustering

Let us consider a splitting of $X \subset \mathbb{R}^N$ into $X_1, \ldots, X_k$. We consider the elements of each cluster X_i to be "coded" by some optimal density f_i out of a family $\mathcal{F}_i$.
We are trying to mininimize the cross entropy of the elements of clusters against that density:

$$\mathrm{CEC}(X_1, f_1; \ldots; X_k, f_k) = \sum_{i=1}^{k} p_i \cdot \left(-\ln(p_i) + H^{\times}(X_i \| f_i) \right), \tag{1}$$

where $p_i = \frac{|X_i|}{|X|}$.

Cross entropy clustering

Let us consider a splitting of $X \subset \mathbb{R}^N$ into $X_1, \ldots, X_k$. We consider the elements of each cluster X_i to be "coded" by some optimal density f_i out of a family $\mathcal{F}_i$.
We are trying to mininimize the cross entropy of the elements of clusters against that density:

$$\mathrm{CEC}(X_1, f_1; \ldots; X_k, f_k) = \sum_{i=1}^{k} p_i \cdot \left(-\ln(p_i) + H^{\times}(X_i \| f_i)\right), \tag{1}$$

where $p_i = \frac{|X_i|}{|X|}$.
What is obtained is the mean code length of a random element of X if using the optimal codes for densities f_i to encode elements of X_i.

Cross entropy clustering

- start with an initial clustering
- for each point $p \in X$:
  - try assigning p to each cluster and calculate CEC
  - choose the cluster with minimal CEC and assign p to it
- repeat until no changes were made during an iteration

Cross entropy clustering

- start with an initial clustering
- for each point $p \in X$:
  - try assigning p to each cluster and calculate CEC
  - choose the cluster with minimal CEC and assign p to it
- repeat until no changes were made during an iteration

The most common density class are gaussian, for which cross entropy is:

$$H^{\times}(\mathrm{y}\|\mathcal{G}_{\Sigma}) = \frac{N}{2}\ln(2\pi) + \frac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma_X) + \frac{1}{2}\ln\det(\Sigma)$$

Table of Contents

Motivation

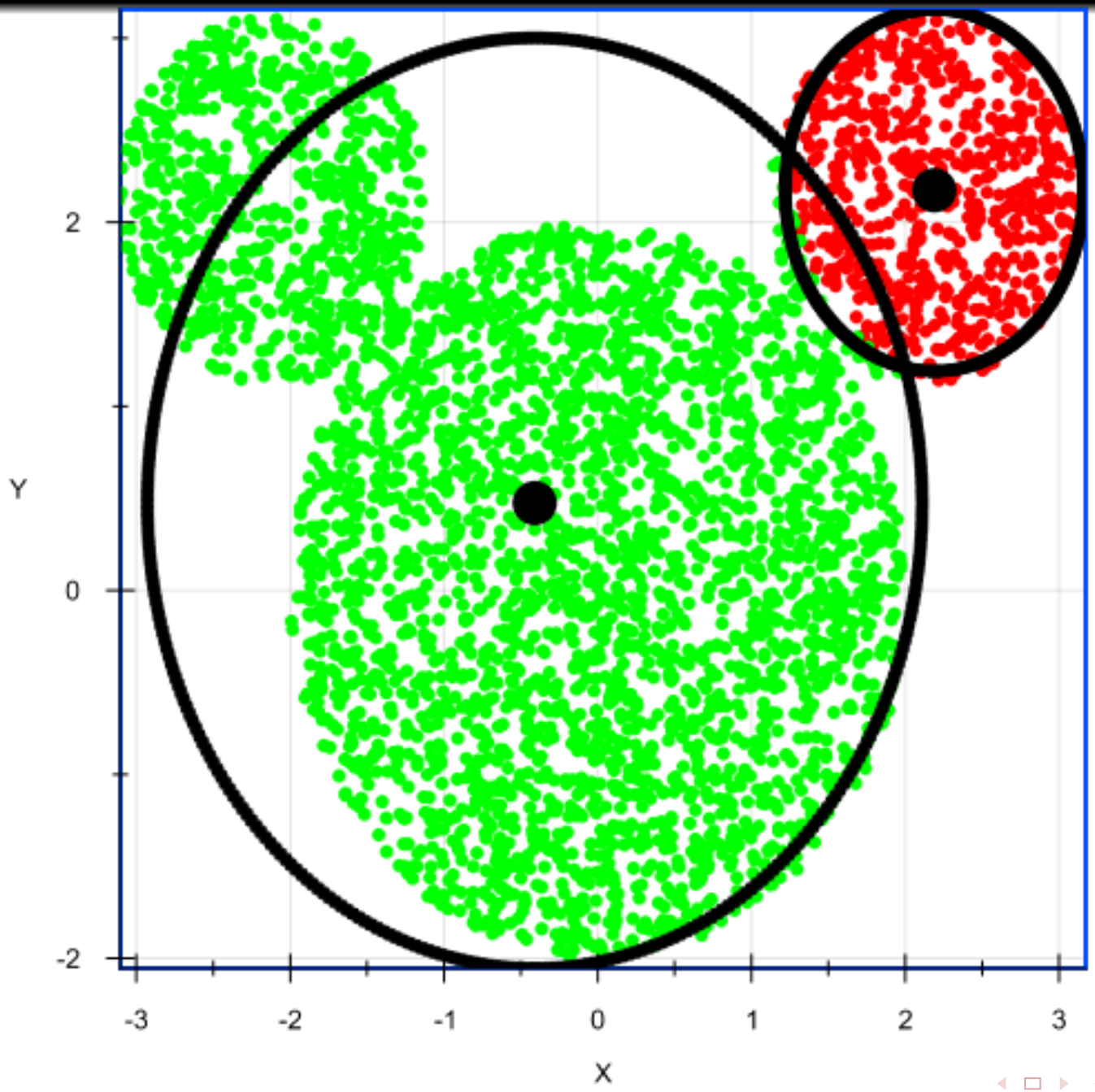
In many applications, the number of clusters is not known

Motivation

In many applications, the number of clusters is not known
Even if the number of clusters is known, the result is very dependent on the initial clustering.

Motivation

In many applications, the number of clusters is not known
Even if the number of clusters is known, the result is very dependent on the initial clustering.
Allowing the algorithm to adjust the number of clusters makes the process more flexible.


2
0
-2
Y
-3
-2
-1
0
1
2
3
X

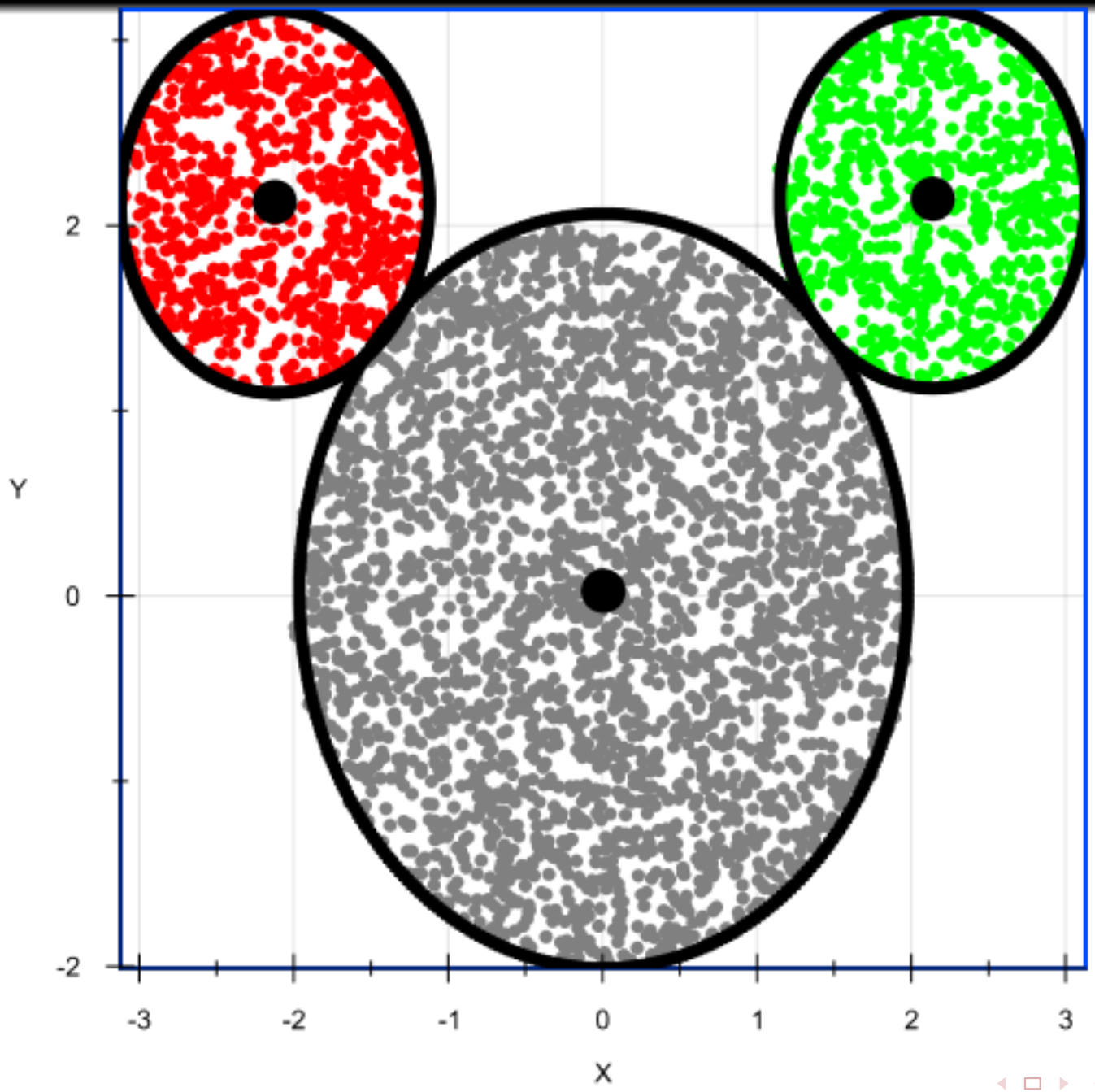

2
0
-2
Y
-3
-2
-1
0
1
2
3
X

Table of Contents

Splitting

If the number of clusters is too small or if the initial set of clusters is unfortunate, one cluster will contain much more elements than would be desired. If the cluster were to be divided into two, the cost of the clustering would reduce.

Splitting

If the number of clusters is too small or if the initial set of clusters is unfortunate, one cluster will contain much more elements than would be desired. If the cluster were to be divided into two, the cost of the clustering would reduce. The simpliest way to find if that is the case, is to run the CEC algorithm on the elements of the clustering, expecting 2 clusters, and see if the cost of the obtained clusters is smaller than that of the original cluster.

Splitting

If the number of clusters is too small or if the initial set of clusters is unfortunate, one cluster will contain much more elements than would be desired. If the cluster were to be divided into two, the cost of the clustering would reduce.
The simpliest way to find if that is the case, is to run the CEC algorithm on the elements of the clustering, expecting 2 clusters, and see if the cost of the obtained clusters is smaller than that of the original cluster.
By applying the above to each of the clusters, we can add clusters where it would be beneficial.

Splitting

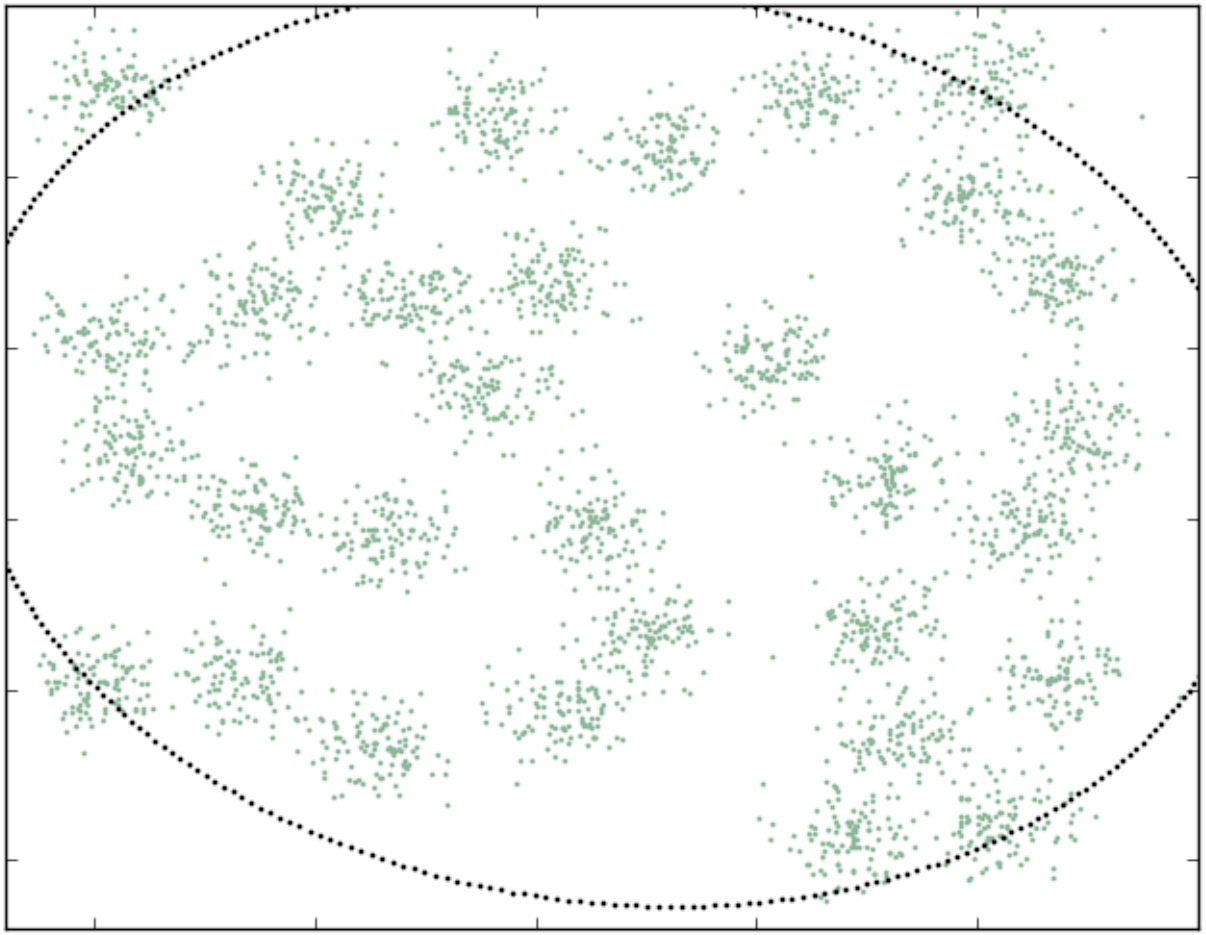
While the many splittings could result in a surplus of clusters, CEC naturally empties unwanted clusters.

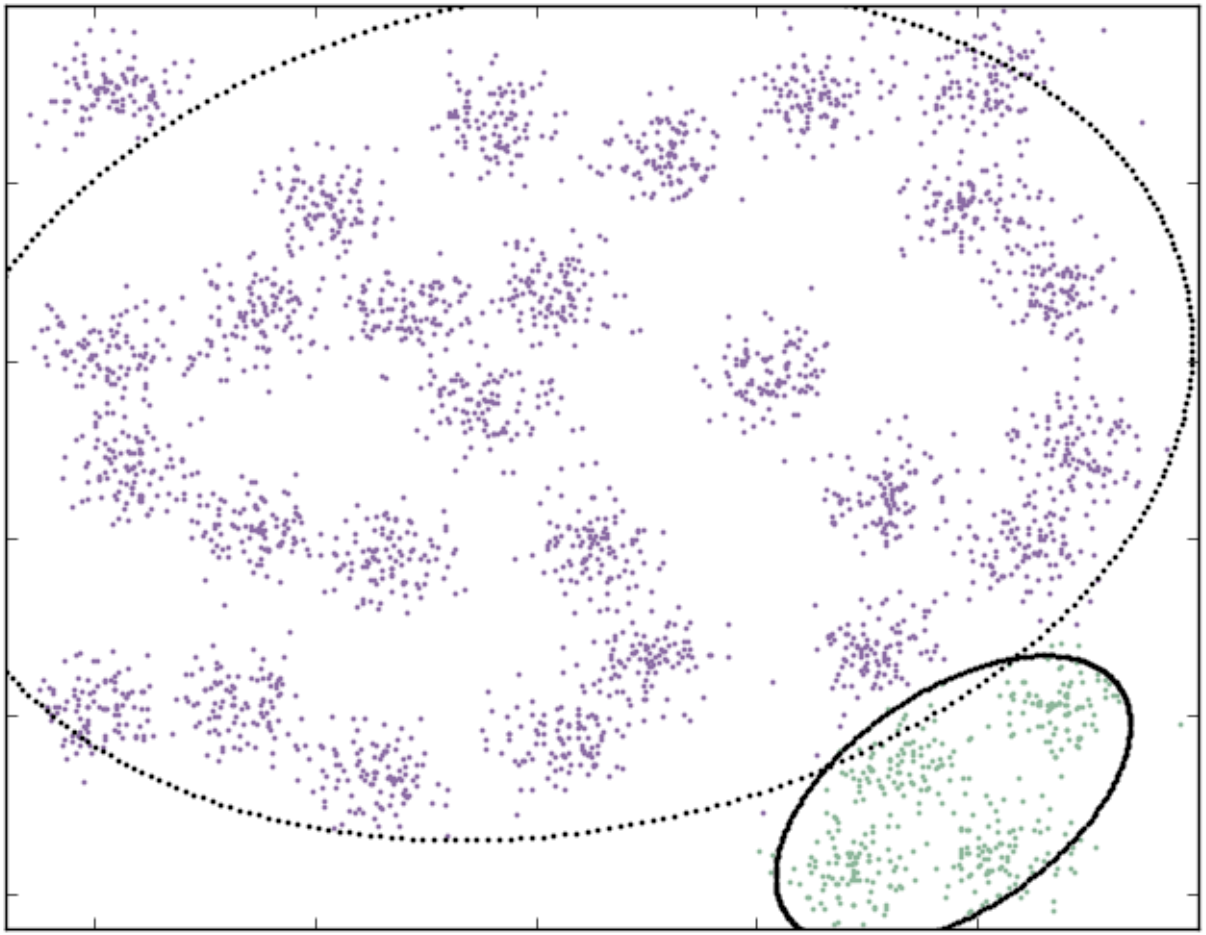
Splitting

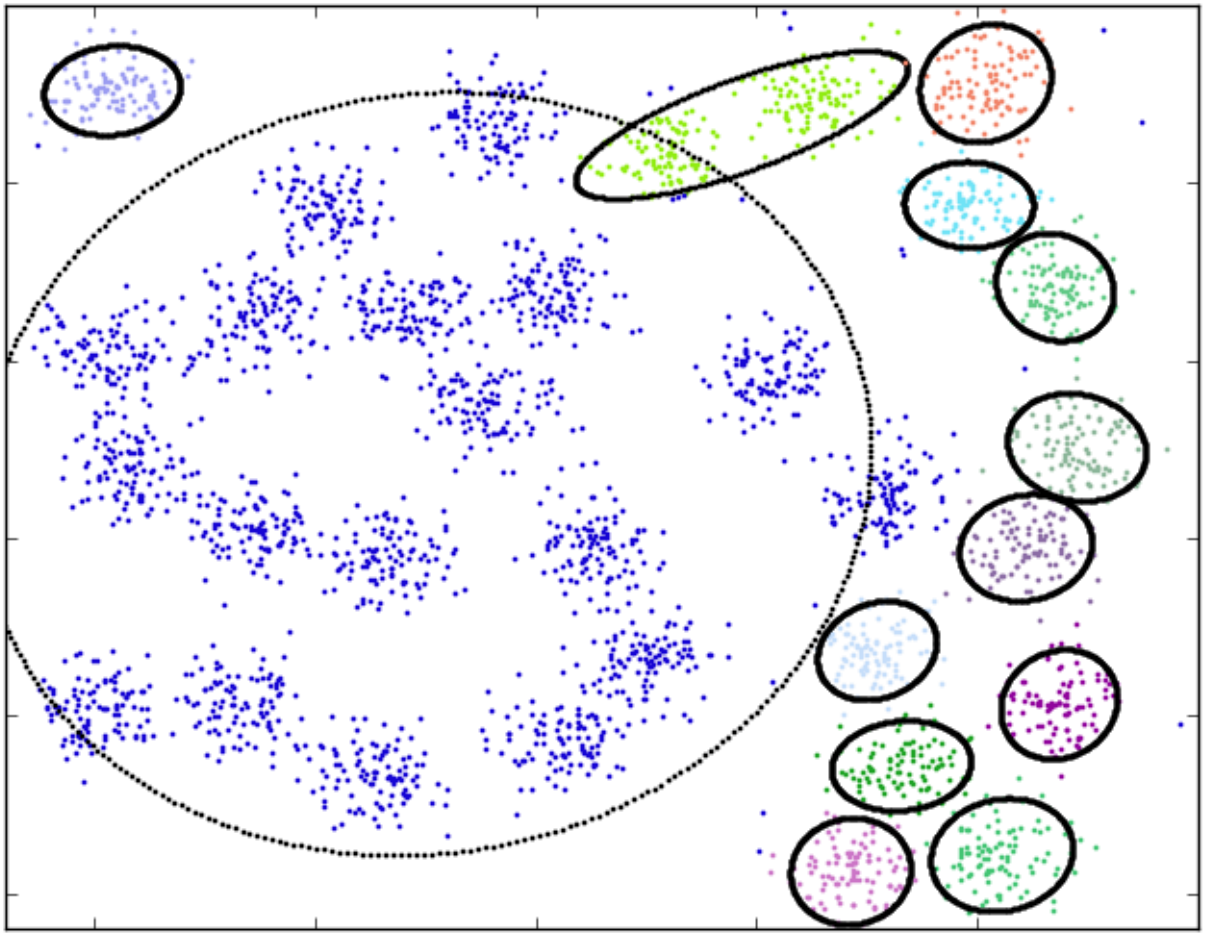
While the many splittings could result in a surplus of clusters, CEC naturally empties unwanted clusters.
Because the number of clusters is penalized in the clustering cost, the splitting will not always be a good option.

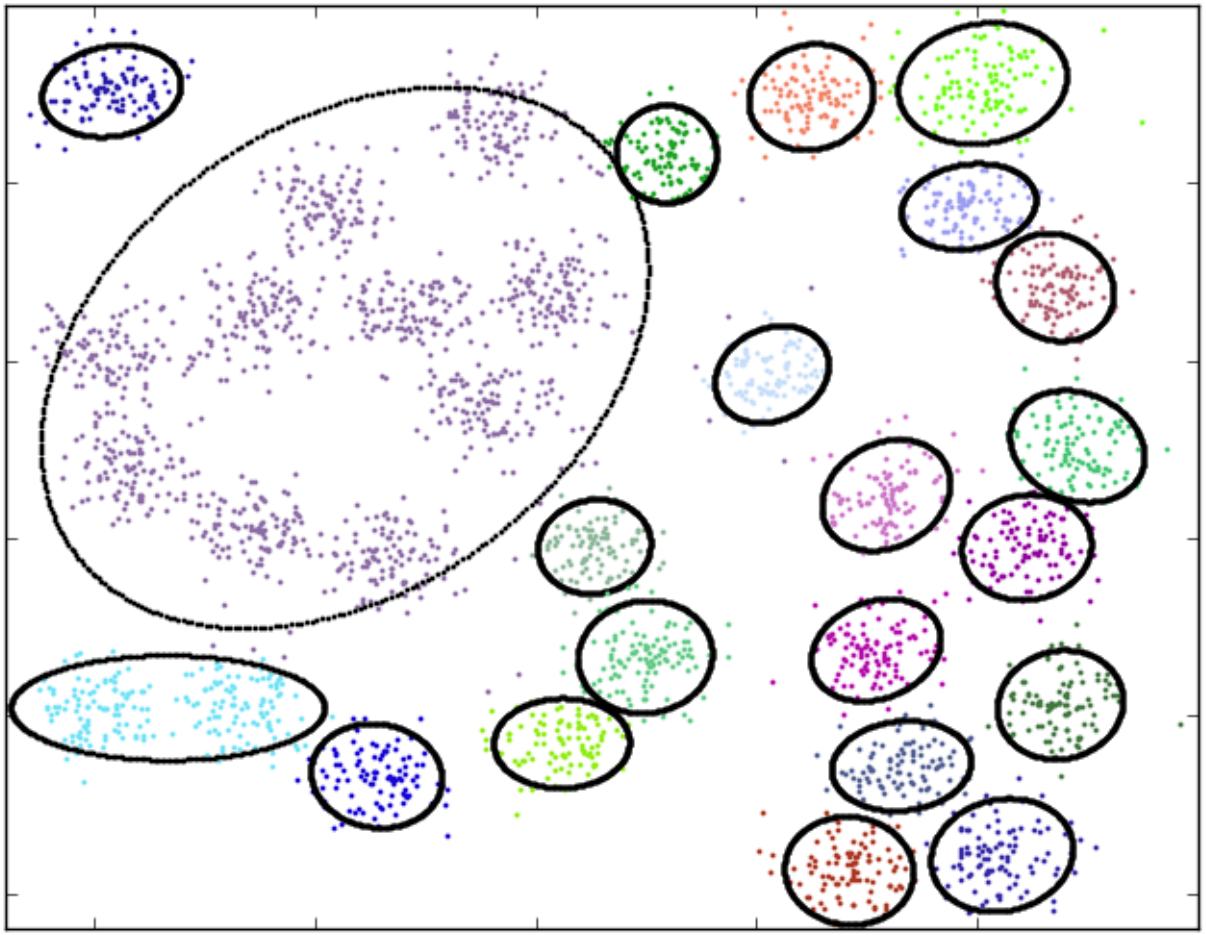
Splitting

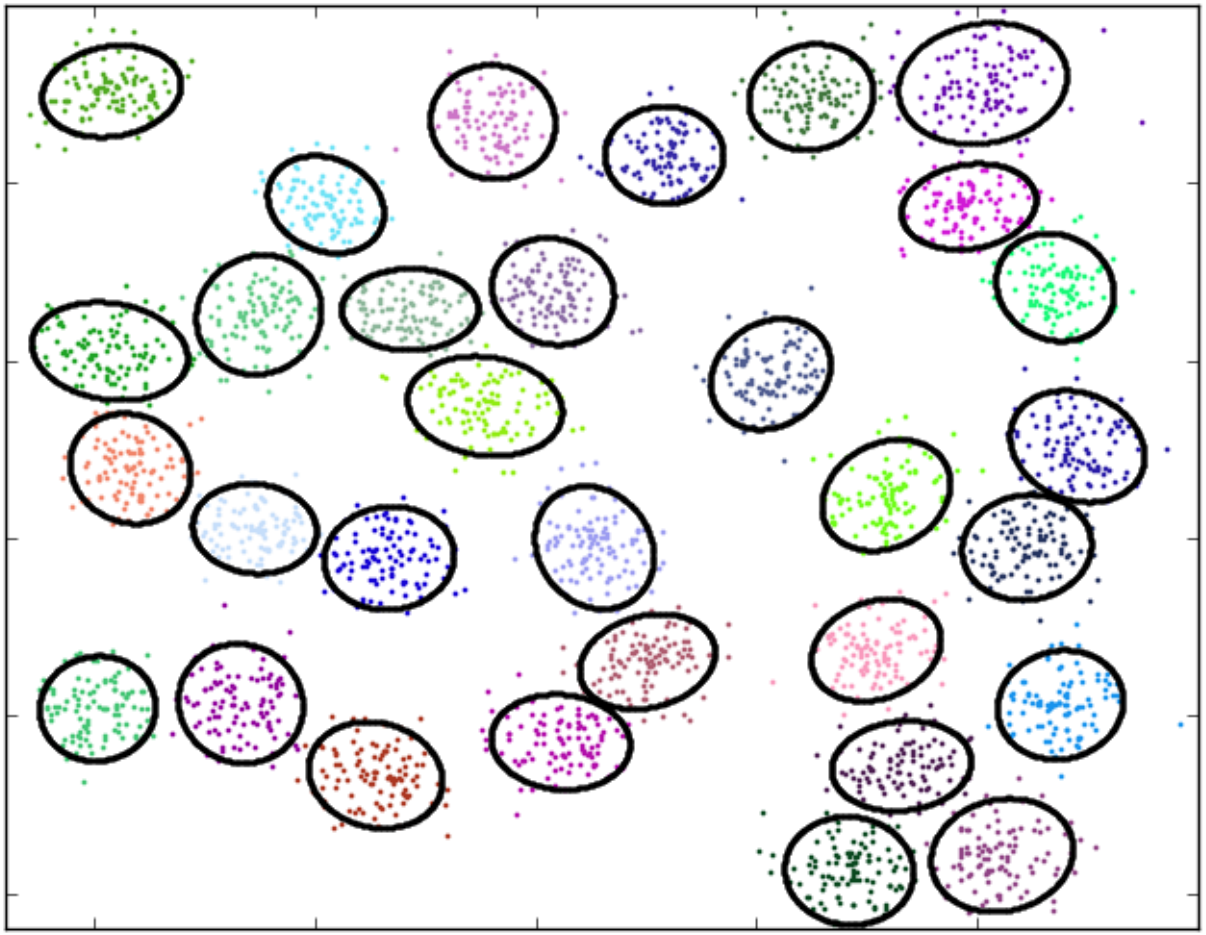
While the many splittings could result in a surplus of clusters, CEC naturally empties unwanted clusters.
Because the number of clusters is penalized in the clustering cost, the splitting will not always be a good option.
With the splitting mechanic, one could start with a single cluster and build from there, obtaining an almost deterministic algorithm.











Merging

Even though CEC empties redundant clusters on its own, it is a slow process.

Merging

Even though CEC empties redundant clusters on its own, it is a slow process.
To make it faster, we apply a similar approach to before: take two clusters, try merging them and see if the clustering cost is reduced.

Merging

Even though CEC empties redundant clusters on its own, it is a slow process.
To make it faster, we apply a similar approach to before: take two clusters, try merging them and see if the clustering cost is reduced.
The merging tweak doesn't provide as much improvement by itself, it works better as a supplement to the splitting tweak.

Merging

Even though CEC empties redundant clusters on its own, it is a slow process.
To make it faster, we apply a similar approach to before: take two clusters, try merging them and see if the clustering cost is reduced.
The merging tweak doesn't provide as much improvement by itself, it works better as a supplement to the splitting tweak.
While one could, similarilly, start with a high number of clusters and consequently merge them, that approach seems to be inefficient for CEC.

Table of Contents

Strategies

- CEC-FAF - check for merging when CEC coverges, repeat as much merges as possible, check for splitting when CEC converges.
- CEC-FSF - as above, but perform only one merge in a single iteration
- CEC-P3AF - every 3 iterations try merging as much as possible, check for splitting when CEC converges
- CEC-P1SP1 - check for merging or splitting after each iteration, at most one merge per iteration

Strategies

- CEC-FAF - check for merging when CEC coverges, repeat as much merges as possible, check for splitting when CEC converges.
- CEC-FSF - as above, but perform only one merge in a single iteration
- CEC-P3AF - every 3 iterations try merging as much as possible, check for splitting when CEC converges
- CEC-P1SP1 - check for merging or splitting after each iteration, at most one merge per iteration

The methods we compared to were grid-search versions of CEC and GMM.

Methodology

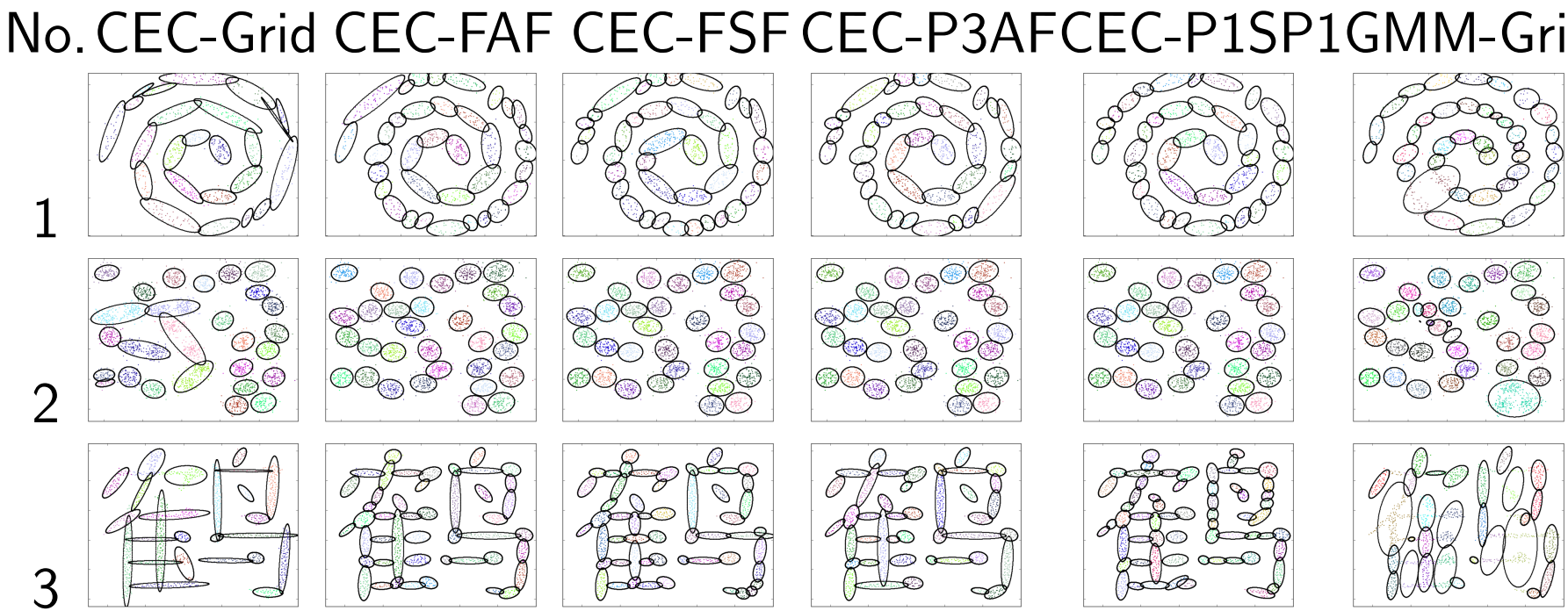
The datasets that were used composed of both gaussian and non-gaussian sets.

Methodology

The datasets that were used composed of both gaussian and non-gaussian sets.
Higher dimensional data was also tested, but not present here.

Methodology

The datasets that were used composed of both gaussian and non-gaussian sets.
Higher dimensional data was also tested, but not present here.
The measures used were maximum loglikelihood, AIC and BIC


No.
CEC-Grid
CEC-FAF
CEC-FSF
CEC-P3AF
CEC-P1SP1
GMM-Grid
1
2
3

Set	Method	MLE	AIC	BIC	cost	No. clust
1	CEC-Grid	-5302.07	**10933.03**	11523.40	5.337437	24
	CEC-FAF	-5296.87	10949.07	11511.24	5.545499	35
	CEC-FSF	-5297.57	10954.84	11516.37	5.544520	38
	CEC-P3AF	-5295.65	10953.27	11503.95	5.546854	36
	CEC-P1SP1	**-5293.99**	10943.12	**11494.34**	5.546223	38
	GMM-Grid	-5476.76	11275.40	12065.55		40
2	CEC-Grid	**-17345.28**	35275.38	36420.90	5.626151	28
	CEC-FAF	-17469.87	**35247.75**	**36177.78**	5.791727	31
	CEC-FSF	-17469.87	**35247.75**	**36177.78**	5.791727	31
	CEC-P3AF	-17469.87	**35247.75**	**36177.78**	5.791727	31
	CEC-P1SP1	-17469.87	**35247.75**	**36177.78**	5.791727	31
	GMM-Grid	-17857.81	35933.63	36591.90		27
3	CEC-Grid	1585.14	-2822.28	**-2088.01**	-1.00290	25
	CEC-FAF	1762.23	-2876.52	-1554.79	-1.09039	46
	CEC-FSF	1738.68	-2876.18	-1566.58	-0.96549	54
	CEC-P3AF	1758.68	**-2893.73**	-1658.30	-1.19903	42
	CEC-P1SP1	**1790.03**	-2855.14	-1260.18	-0.85854	63
	GMM-Grid	815.81	-1413.62	-834.12		27

Future work

Instead of explicitely performing the splitting and merging, we could use some measure to decide wether we should perform the split or merge.

Future work

Instead of explicitely performing the splitting and merging, we could use some measure to decide wether we should perform the split or merge.
The method does not have any element specific to CEC, so it could be considered for other clustering algorithms as well.

Thank you for your attention.