

## Data analysis with the BFR clustering algorithm.

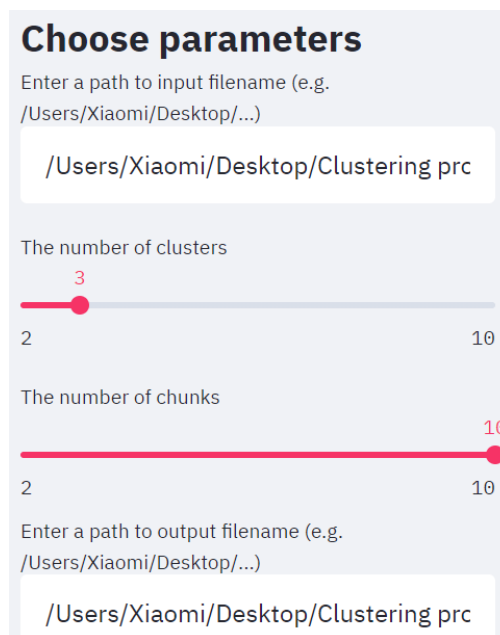
This documentation introduces the usage of the BFR algorithm Web application. The algorithm was developed based on Python programming language using imported libraries such as streamlit, numpy, sklearn and matplotlib.

The main goal of the BFR algorithm is to contribute to analysis of a data set, that is too big to be uploaded in the main memory wholly. It splits a data set into several parts and creates clusters based on Kmeans clustering algorithm.

To start clustering process, the needed paths to the files must be given and parameters are to set. On the left side of the web app there is two blank lines where the paths to input file with dataset in csv. or txt. formats and to output file in txt. format, where the results will be written, should be given. The requirements for the data set are:

1. It must contain only numerical values.
2. The first raw with the names of the attributes must be removed (if it was there).
3. The first column with the indices from 0 till n-1 must be added.

User can also choose the number of clusters (from 2 to 10) and the number of parts (from 2 to 10), in which dataset will be divided.



The screenshot shows a web application titled "Choose parameters". It contains two text input fields for file paths, both with the value "/Users/Xiaomi/Desktop/Clustering prc". Between the inputs are two sliders. The first slider is labeled "The number of clusters" and has a value of 3. The second slider is labeled "The number of chunks" and has a value of 10. Both sliders have a range from 2 to 10.

**Choose parameters**

Enter a path to input filename (e.g. /Users/Xiaomi/Desktop/...)

/Users/Xiaomi/Desktop/Clustering prc

The number of clusters

3

2 10

The number of chunks

10

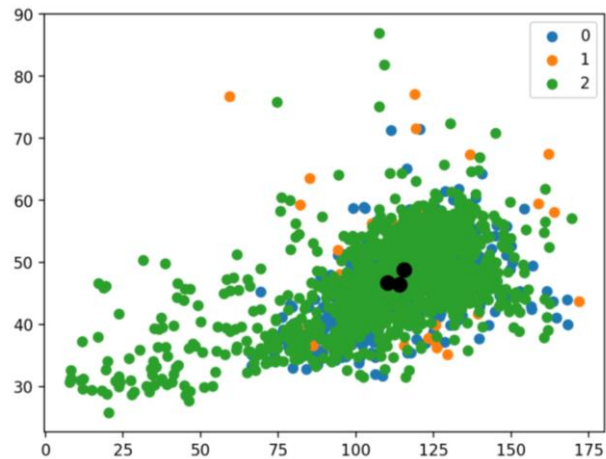
2 10

Enter a path to output filename (e.g. /Users/Xiaomi/Desktop/...)

/Users/Xiaomi/Desktop/Clustering prc

On the left side user will be able to see outputs of the clustering.

1. Clusters, that were produced after the loading first data chunk and applying kmeans function, with the respective centroids are plotted.



- Intermediate results for the neat loads which show the number of points in all general clusters (discard sets), the number of points in small clusters and the number of clusters (compressed sets), the number of points that were not assigned to any clusters (retained set).

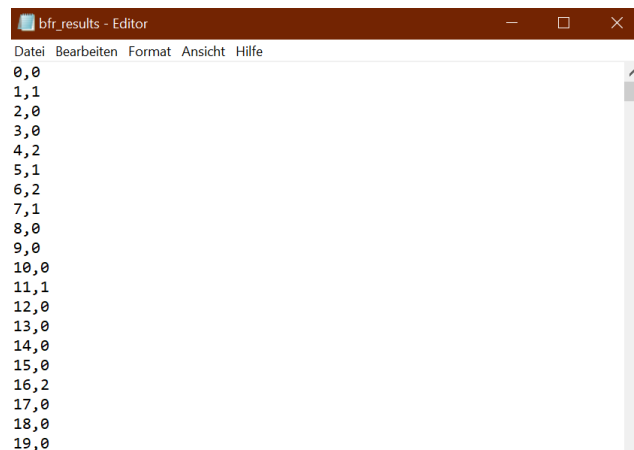
#### The intermediate results:

Round 1: DS Points: 8713,CS Clusters: 14,CS Points: 236,RS Points: 1

Round 2: DS Points: 12941,CS Clusters: 27,CS Points: 481,RS Points: 2

Round 3: DS Points: 17308,CS Clusters: 40,CS Points: 589,RS Points: 1

- Clustering results can be found in the output file as the indexes of the points and number of cluster to which every point was assigned.



- The results of the comparison with the Hierarchical and the Kmeans clustering algorithms which are calculated applied the Rand index.

#### Comparison of the algorithms:

The Rand index (BFR and Hierarchical clustering): 0.4789603160655804

The Rand index (BFR and Kmean clustering): 0.4270812569576828