Eötvös Loránd Tudományegyetem

# Statistical Data Generation Using Sample Data

Eötvös Loránd University
Faculty of Informatics
Department of Information Systems
Budapest, Hungary

Bálint Fazekas bfazekas@inf.elte.hu
Attila Kiss kissae@ujs.sk

# Motivations

- Pragmatic and truthful database generation
- Using small sample to work with
- Type and value independent data generation
- Dimension independent
- Data augmentation
- Relatively simple & relatively effective methods
- Legal use of truthful information

Eötvös Loránd
Tudományegyetem

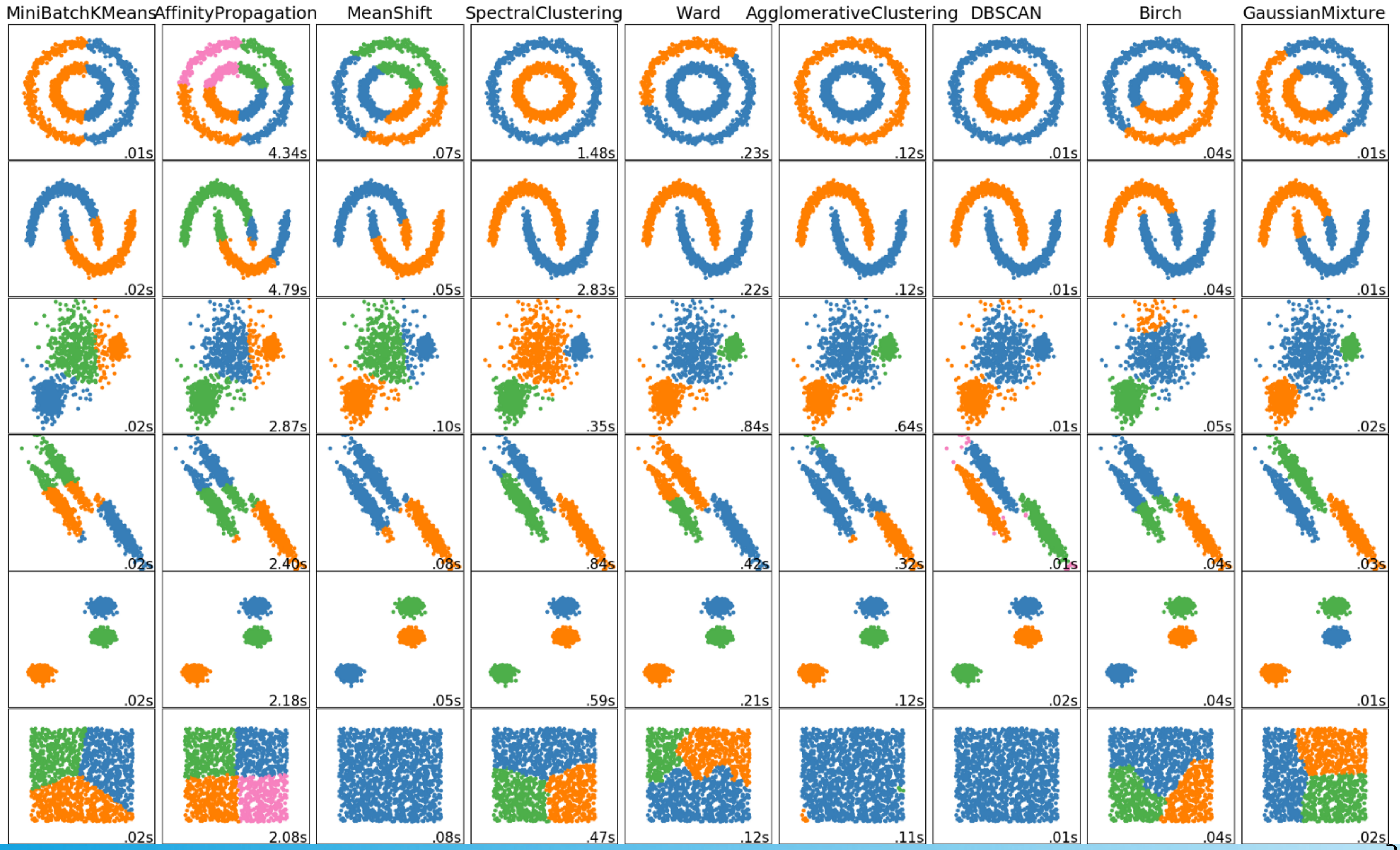
# Numerical representation

- Vectorisation of data
- Not intuitive nor a straightforward task
- Needs heuristic approach based on our desire to observe the given dataset
- Our choice: length based representation of strings

# Data set processing

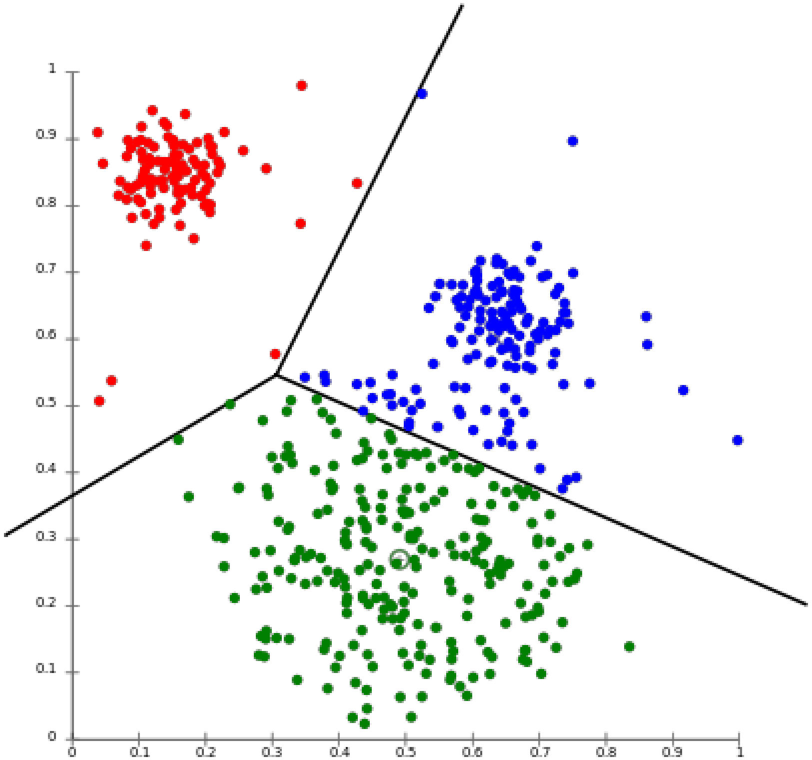
- All data must be represented numerically, in all dimensions
- Statistical analysis and clusterisation can be done on the dataset.

# Clustering methods



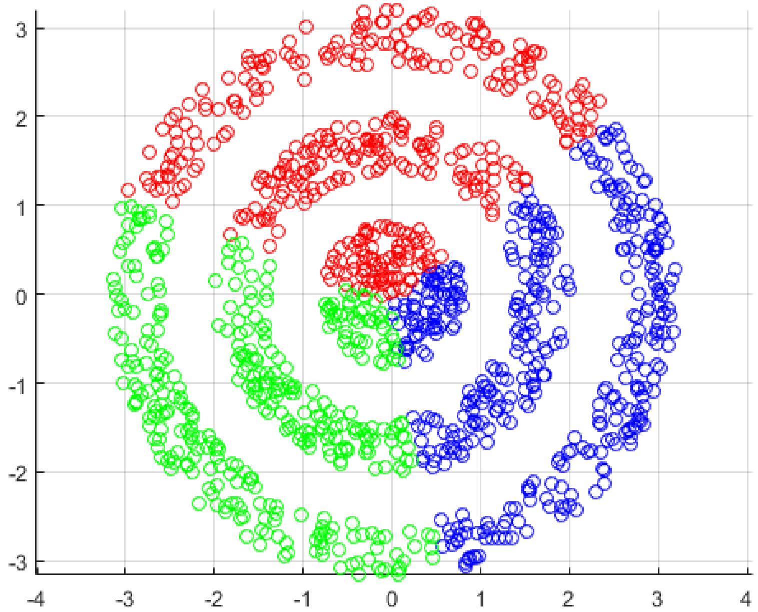
# Clustering methods: K-Means

- Spacio-temporal method
- Non-deterministic!
- Pros:
  - Can identify clusters in any dimensions
  - Does not lose data during clusterisation
  - Fast



# Clustering methods: K-Means

- Cons:
  - Does not give good results in case of concave cluster shapes
  - Outliers can have a great impact on the identified clusters



# Evaluation of K-Means

- Efficient
- Only works intuitively if the datasets have discrete “blobs” of data
- Assumes that the given dataset is spacially distinguishable

- Conclusion: it is not a good clustering method when it is used all by itself

# Clustering methods: DBSCAN

- “Density-based spatial clustering of applications with noise”
- Considers the distances between individual points, rather than observing the dataset as a whole.
- “More intuitive”, therefore the result is more truthful
- Identifies outliers in the dataset.

# Clustering methods: DBSCAN

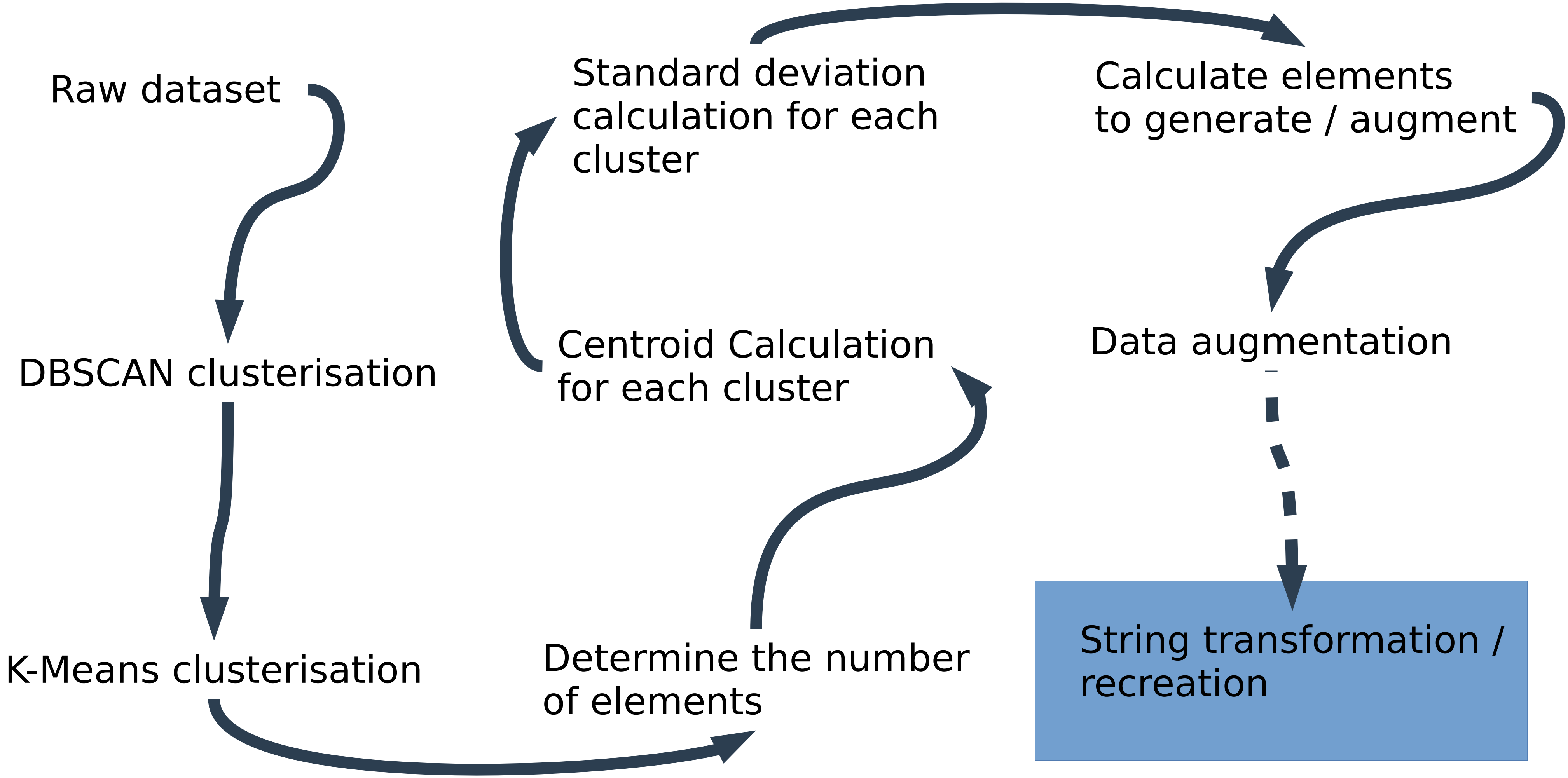
- Cons:
  - When observing the clusters, we can obtain only one centroid
  - Identified <u>concave</u> clusters results in a shifted centroid
  - *$O(n^2)$ very slow*

- Conclusion: the method is adequate for single clustering, but not for the purpose of data generation.

# Hybrid combination

- Due to the time complexity of the DBSCAN algorithm, it should be run the minimum number of times on the entire dataset.
- The result is good clusterization, which is bad for data regeneration.
- Run the K-Means algorithm on each of the clusters created by the DBSCAN algorithm.

Eötvös Loránd Tudományegyetem



Eötvös Loránd Tudományegyetem

# Hybrid algorithm

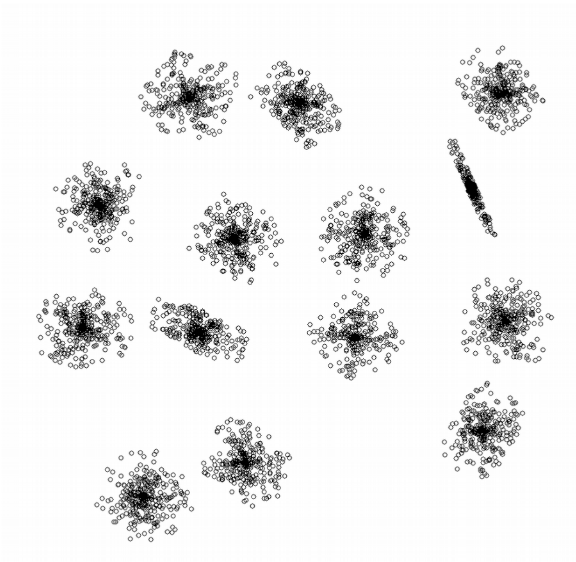
- Exploits the positive characteristics of K-Means and DBSCAN.
- Avoids the weaknesses of both algorithms.
- Creates a number of clusters that can be analyzed statistically and correctly for the purpose of data augmentation / regeneration.

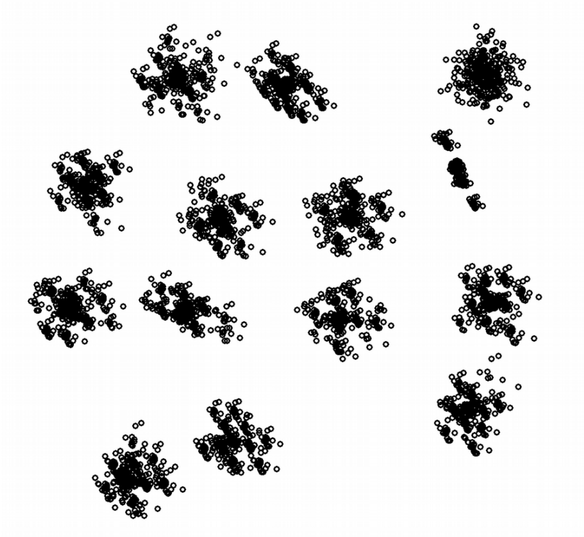
# Statistical analysis

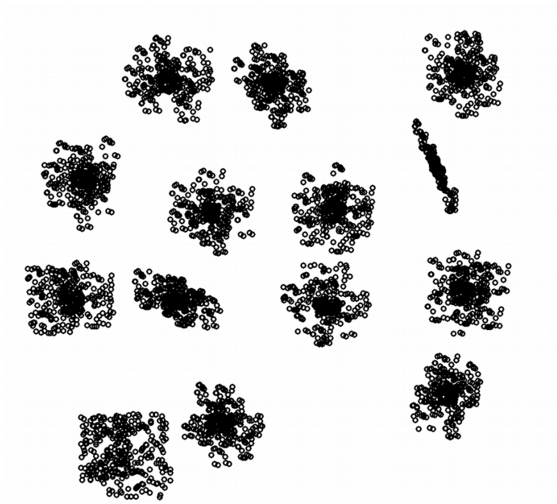
- Finding the multidimensional means (centroids) of all resulting clusters
- Finding the standard deviations of all dimensions of a cluster.
- Generating data by with multivariate normal distribution methods.
- Augmentation: Recreate each cluster with same number of data, multiplied by a factor of *m.*

# Results

System: Ubuntu 17.10
Processor: Intel i7-4710HQ @ 2.50 GHz

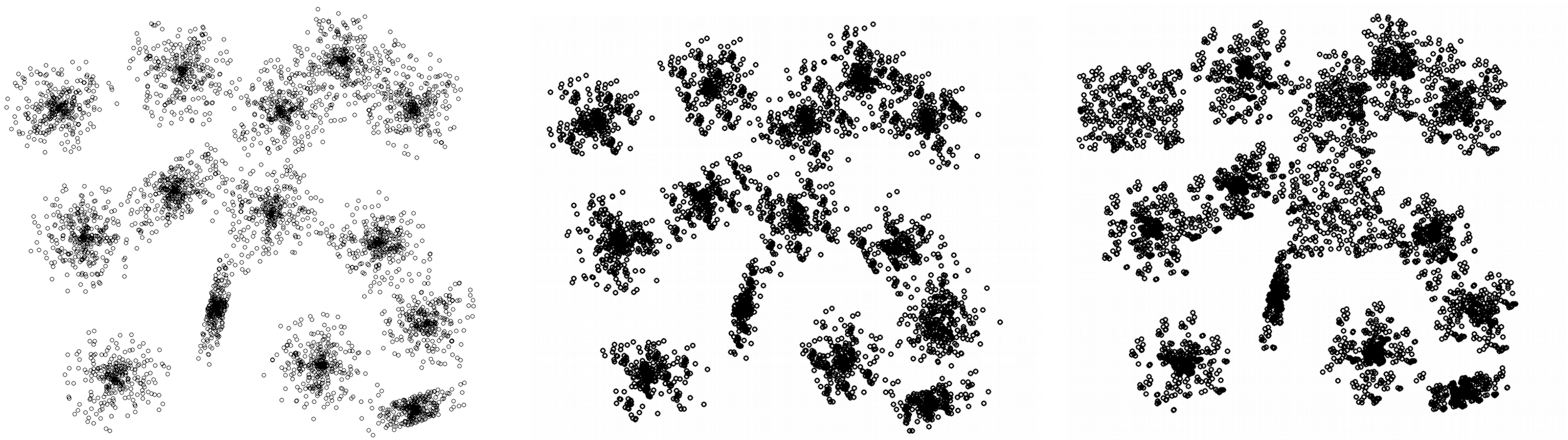






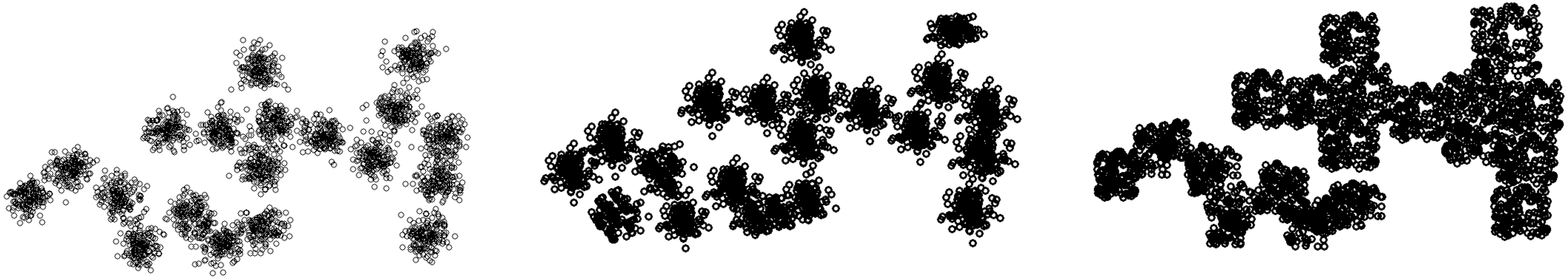
# Results

System: Ubuntu 17.10
Processor: Intel i7-4710HQ @ 2.50 GHz

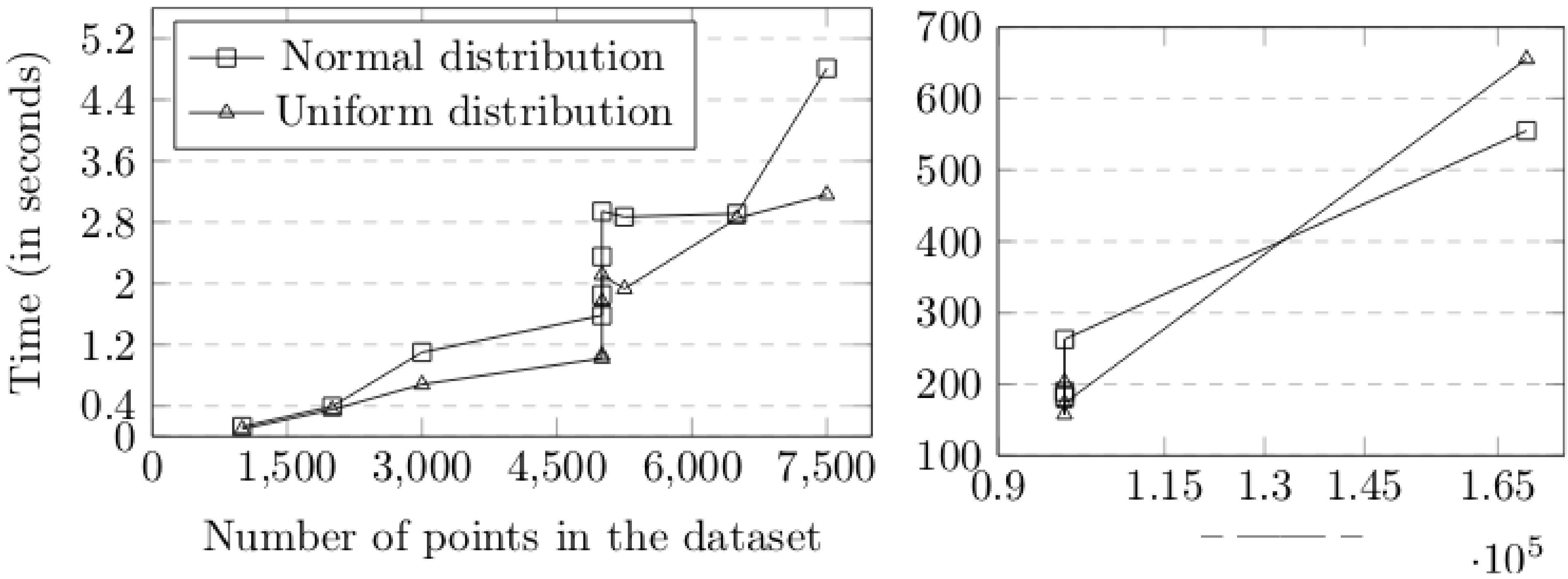


# Results

System: Ubuntu 17.10
Processor: Intel i7-4710HQ @ 2.50 GHz



# Speed


Normal distribution
Uniform distribution
Time (in seconds)
5.2
4.4
3.6
2.8
2
1.2
0.4
0
0
1,500
3,000
4,500
6,000
7,500
Number of points in the dataset
700
600
500
400
300
200
100
0.9
1.15
1.3
1.45
1.65
·10^5

Eötvös Loránd Tudományegyetem

# Accuracy

| Dim. | Number of clusters | Largest difference in distributions of a corresponding cluster in every dimensions |
|---|---|---|
| 2 | 11 | 2.375773, 2.740710<br>2.185325, 2.278970 |
| 3 | 12 | 32.206703, 40.238171, 37.005211<br>32.355637, 31.006762, 34.254261 |
| 5 | 11 | 50.881187, 53.762848, 61.116722, 67.602982, 57.316814<br>52.398487, 50.678226, 47.839535, 53.783073, 50.094315 |
| 9 | 8 | 57.362629, 62.505939, 59.682968, 55.935524, 61.643772,<br>70.316696, 63.057480, 65.735901, 61.742653<br>60.590473, 58.573429, 53.545685, 57.154350, 57.458927,<br>58.119232, 55.449238, 59.794643, 55.689960 |
| 12 | 13 | 66.344612, 53.064713, 56.163540, 55.857380, 67.622047, 63.862854, 61.600574,<br>62.900955, 62.944111, 66.601341, 62.751911, 63.505623<br>70.387451, 61.990356, 68.966225, 65.002014, 66.271294, 65.187050, 68.951828,<br>65.793198, 61.893456, 64.270515, 63.222546, 69.350632 |

# Improvements

- The DBSCAN algorithm requires user input, a distance threshold. The automation of this is not straightforward. *D' ≈ avg(distances measured from 10%-25% of the data)*
- Different methods of handling string data.
- GP/GPU and Threading usages as possible speedup processes for the K-Means algorithm.
- Regeneration should be done in a constricted space, rather than an n-dimensional box.
- Image data: regenerate only meta-data, witch possible noise image.

Eötvös Loránd Tudományegyetem

# Acknowledgements

This project was supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00002).