FSDM
Conference on Fuzzy Systems and Data Mining

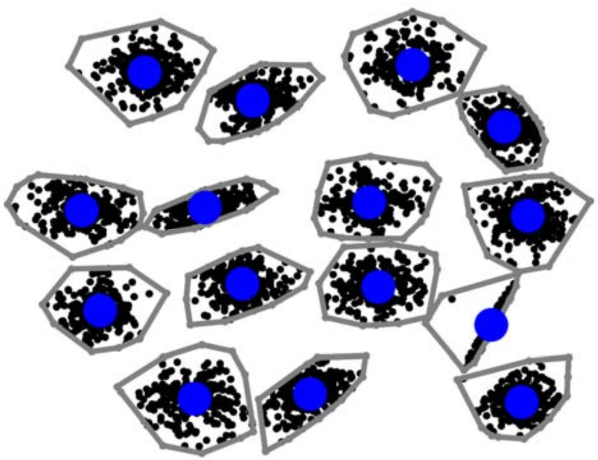
UNIVERSITY OF EASTERN FINLAND

# Mean-shift outlier detection

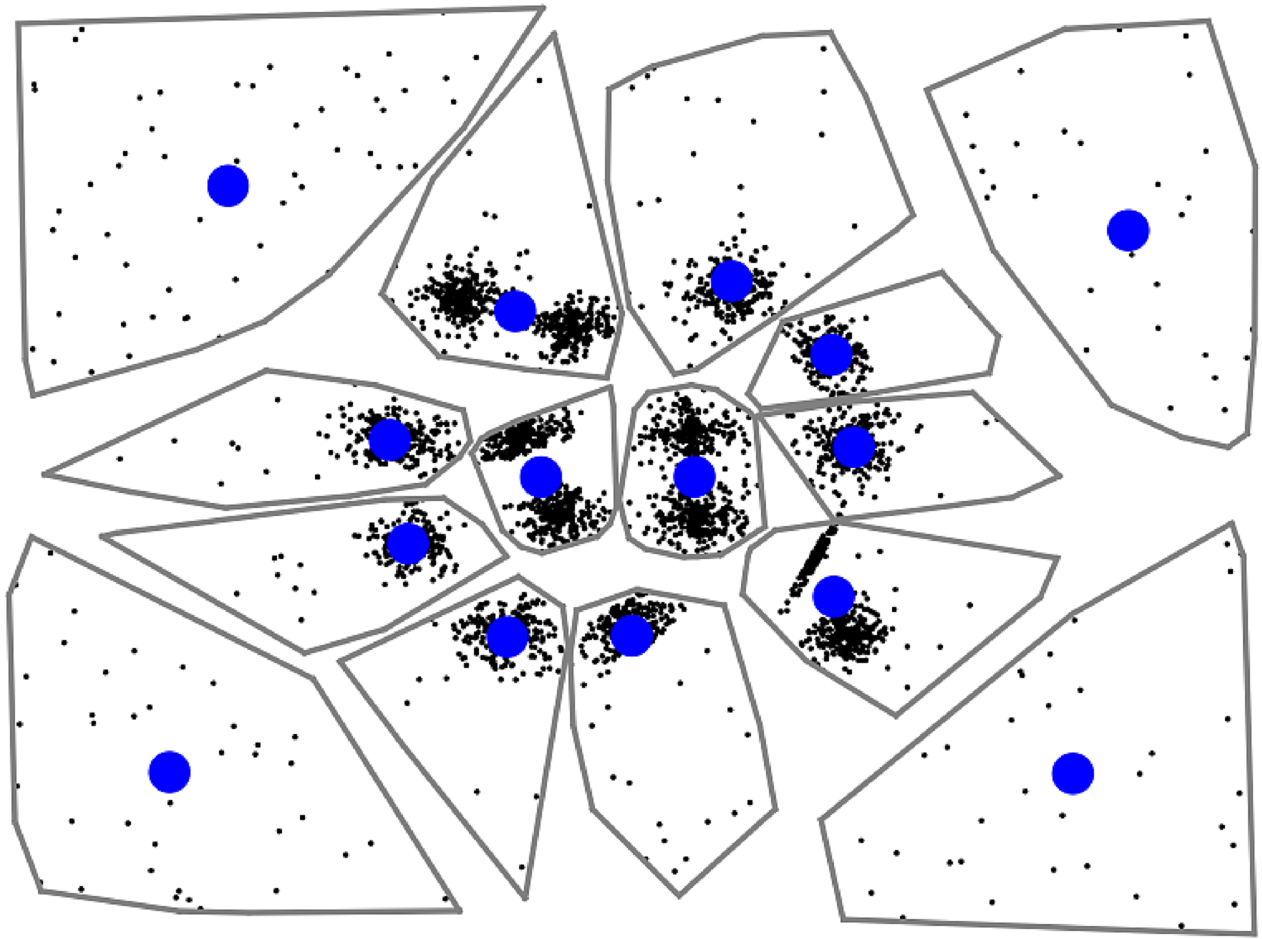
Jiawei Yang
Susanto Rahardja
Pasi Fränti

18.11.2018

# Clustering

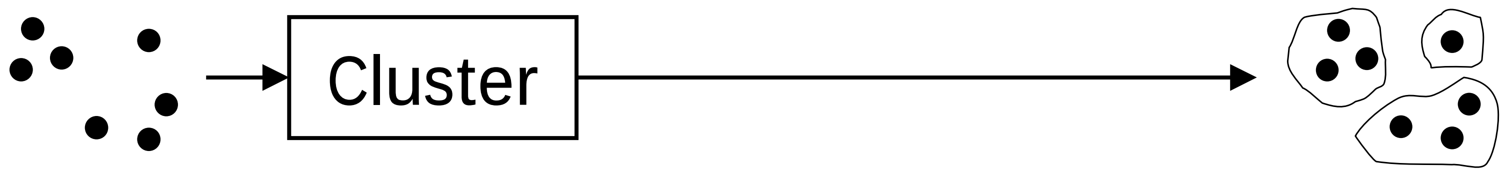


# Clustering with noisy data

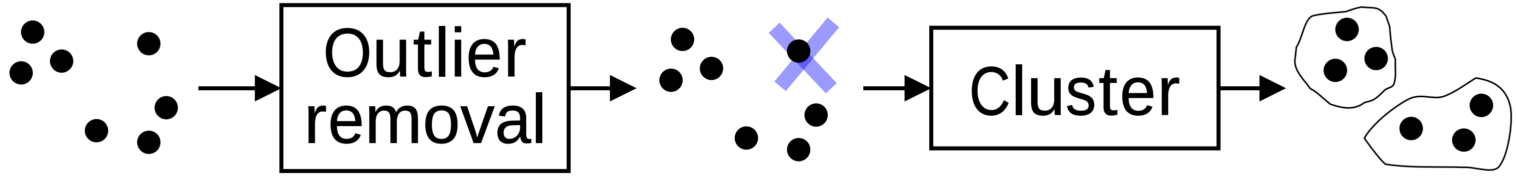


# How to deal with outliers in clustering

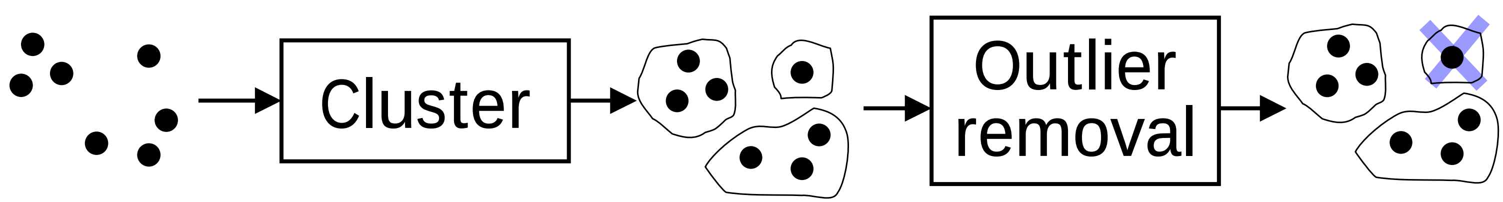
**Approch 1:**



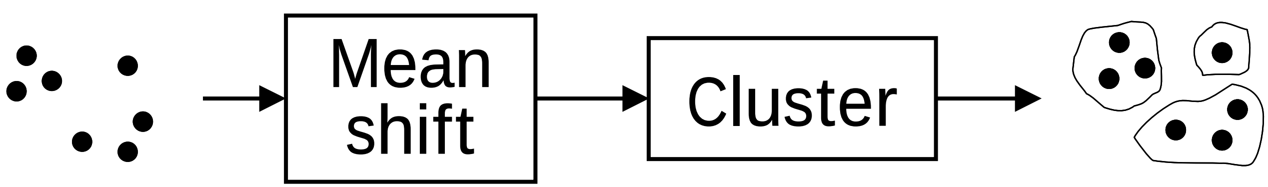

**Approch 2:**




**Approch 3:**



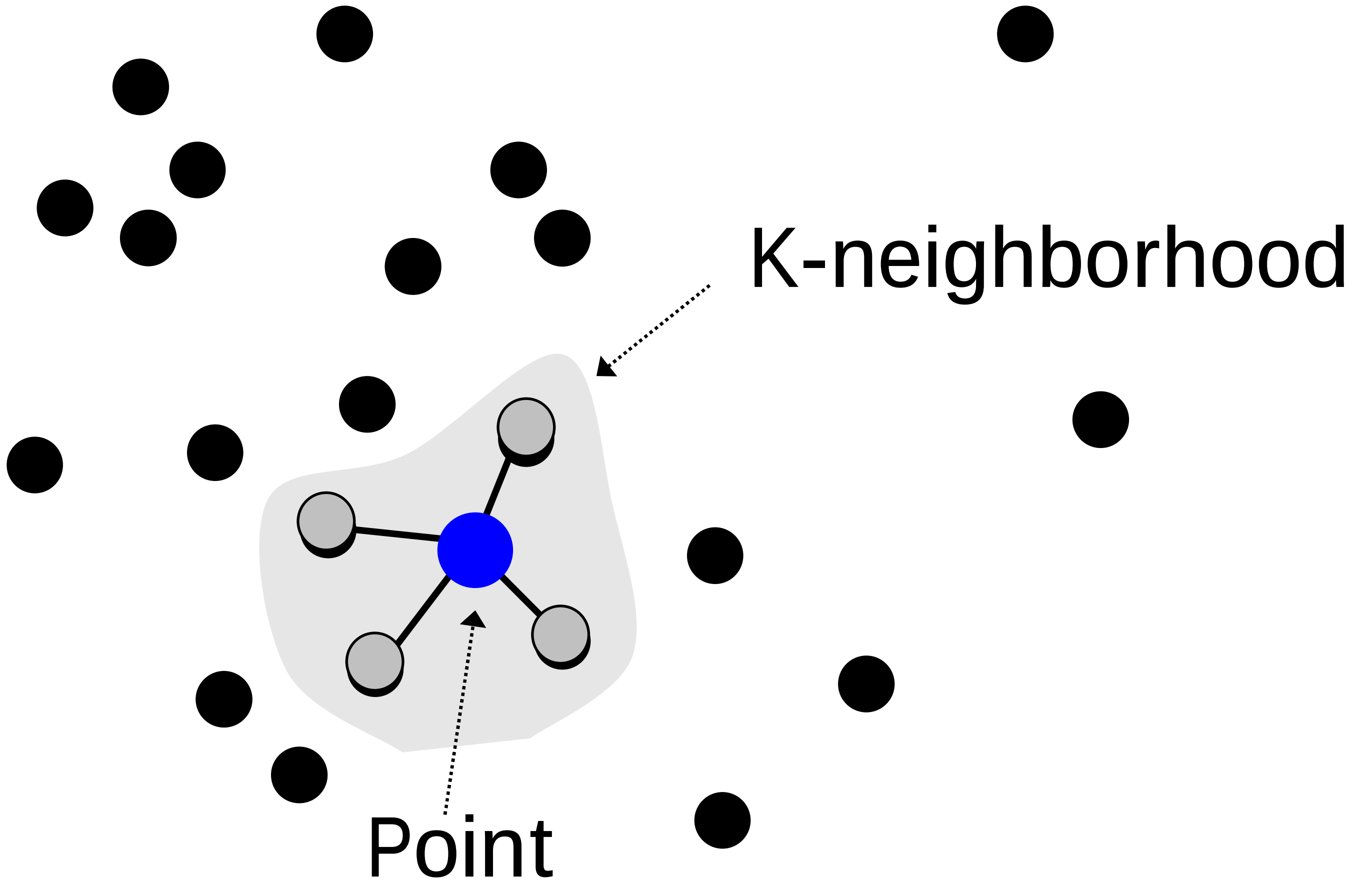

**Mean-shift approch:**



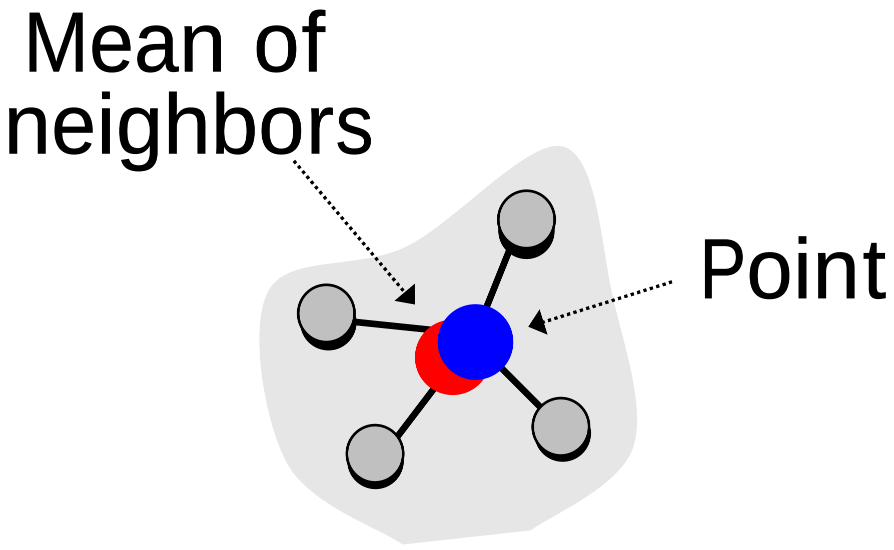
# Mean-shift

# K-nearest neighbors

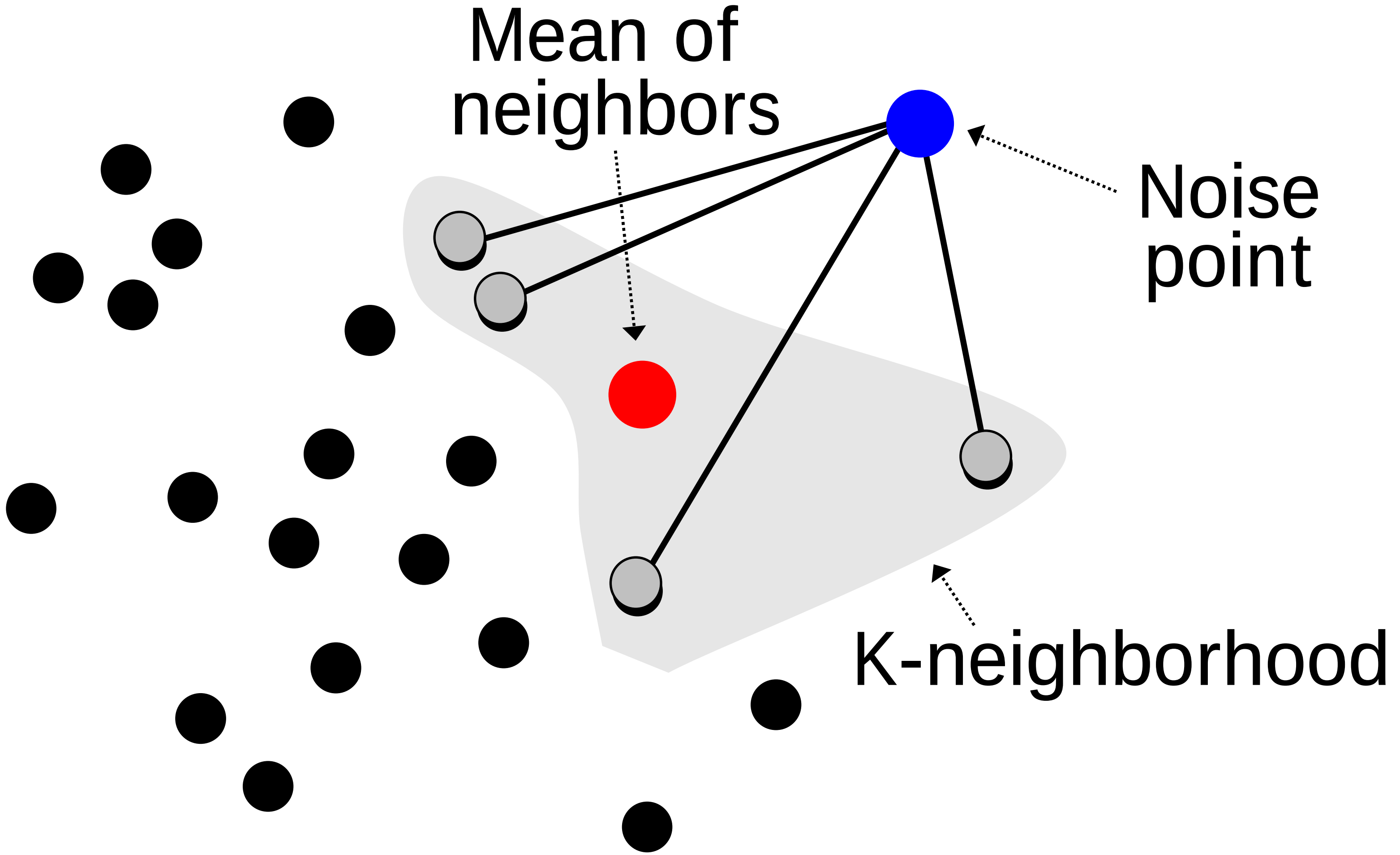
k=4



# Expected result



# Result with noise point

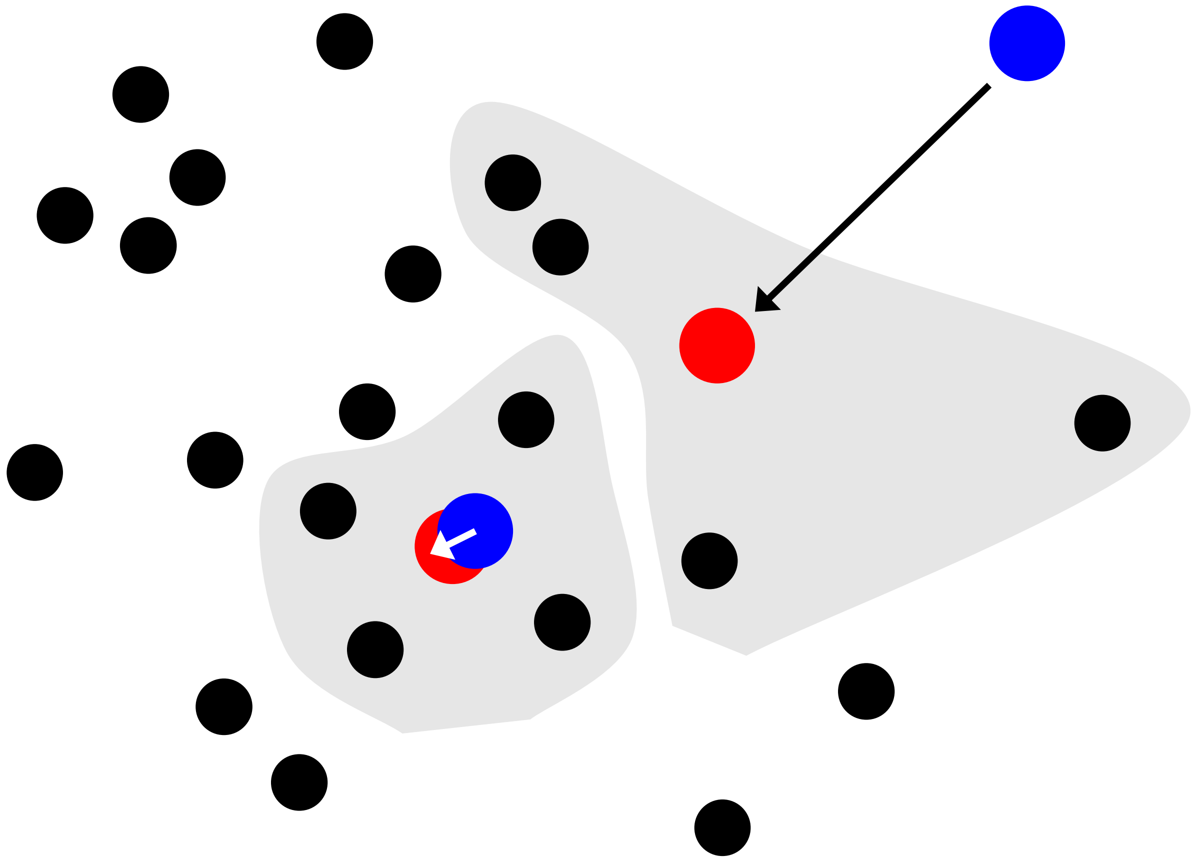


# Part I:
# Noise removal

Fränti and Yang, "Medoid-shift noise removal to improve clustering",
*Int. Conf. Artificial Intelligence and Soft Computing (CAISC)*, June 2018.

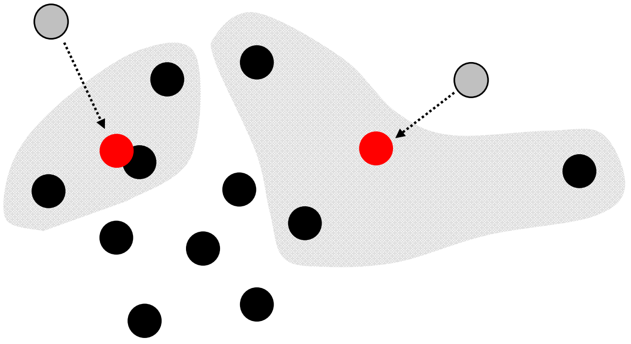
# Mean-shift process

Move points to the means of their neighbors

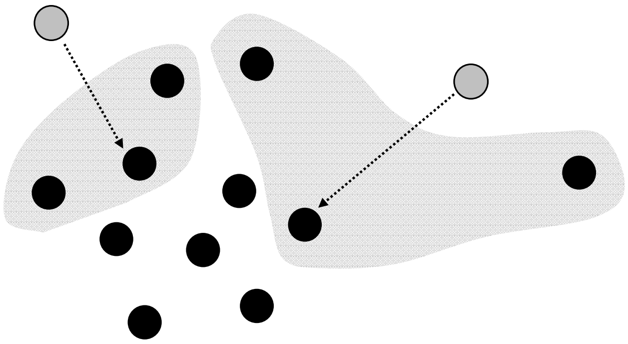


# Mean or Medoid?

**Mean-shift**



**Medoid-shift**



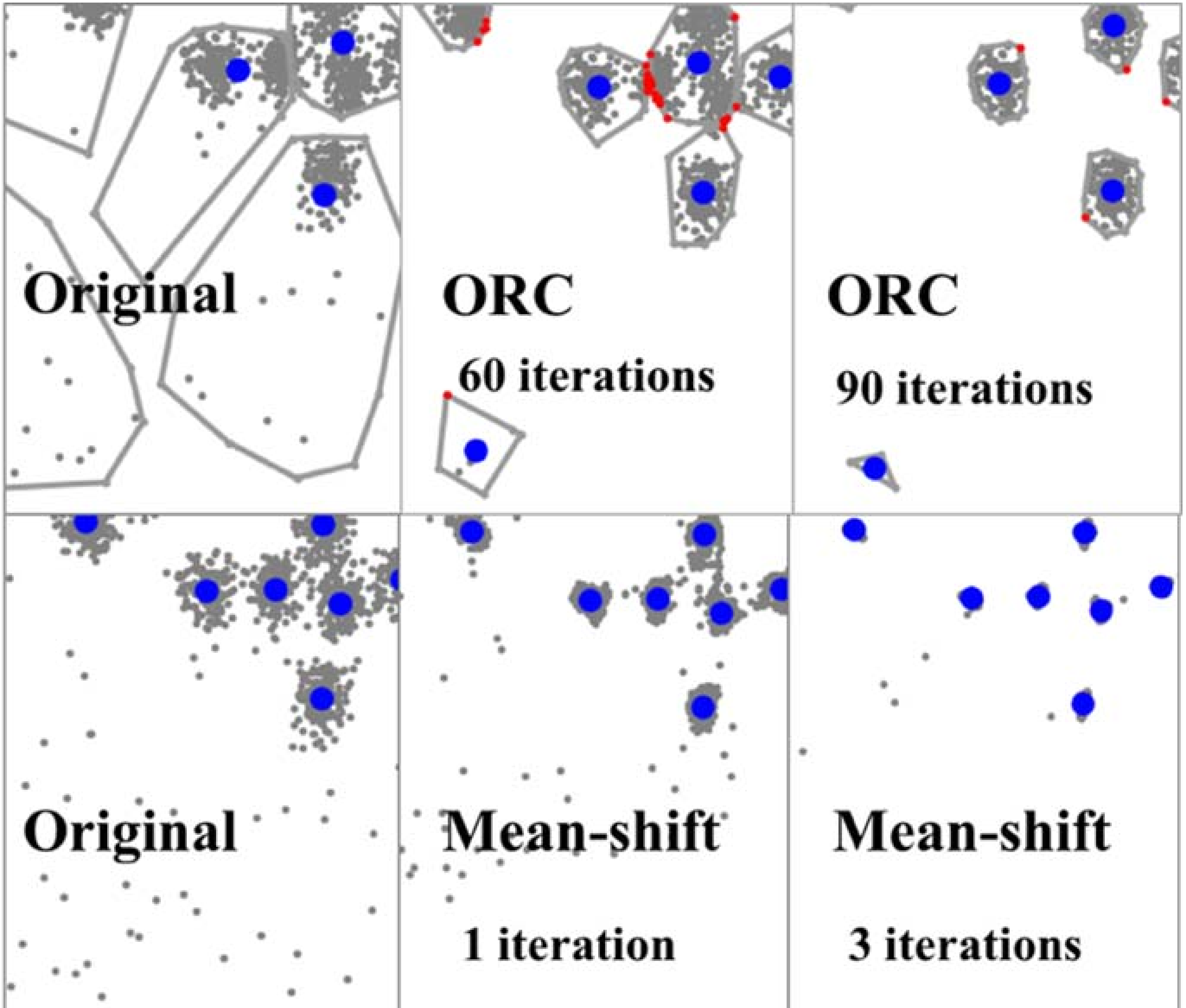
# Medoid-shift algorithm

Fränti and Yang, "Medoid-shift noise removal to improve clustering",
*Int. Conf. Artificial Intelligence and Soft Computing (CAISC)*, June 2018.

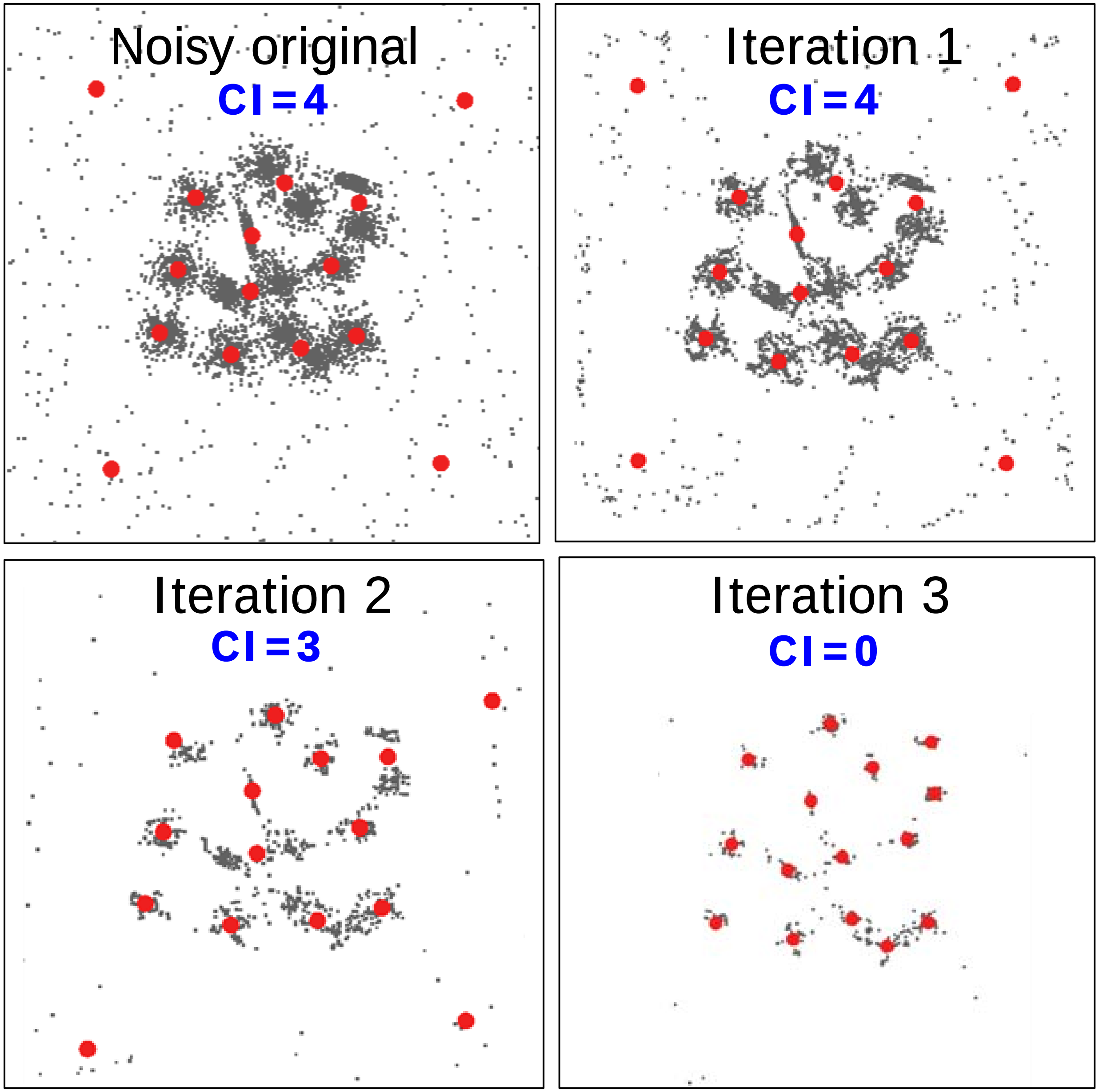
REPEAT **3** TIMES

1. Calculate kNN(x)
2. Calculate medoid M of the neighbors
3. Replace point x by the medoid M

# Iterative processes



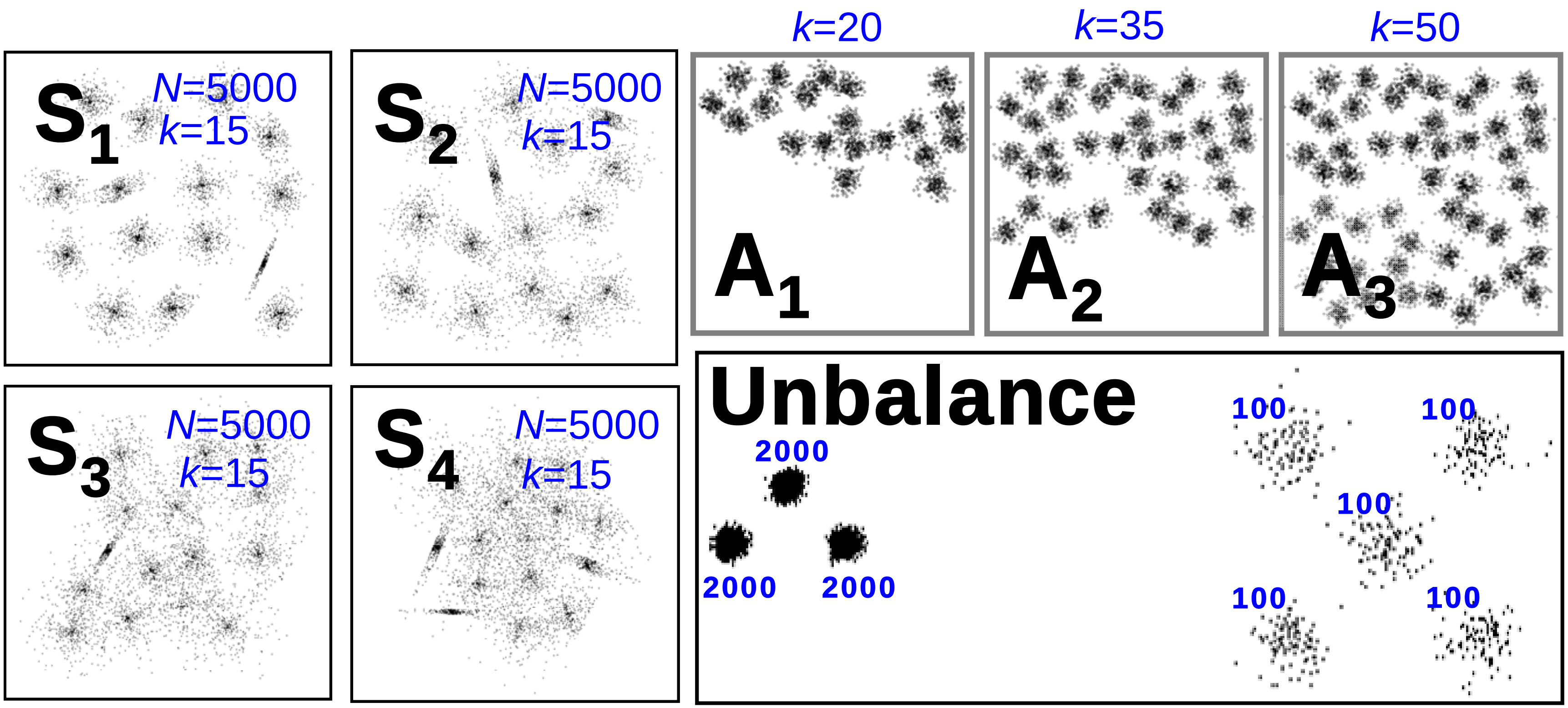
# Effect on clustering result



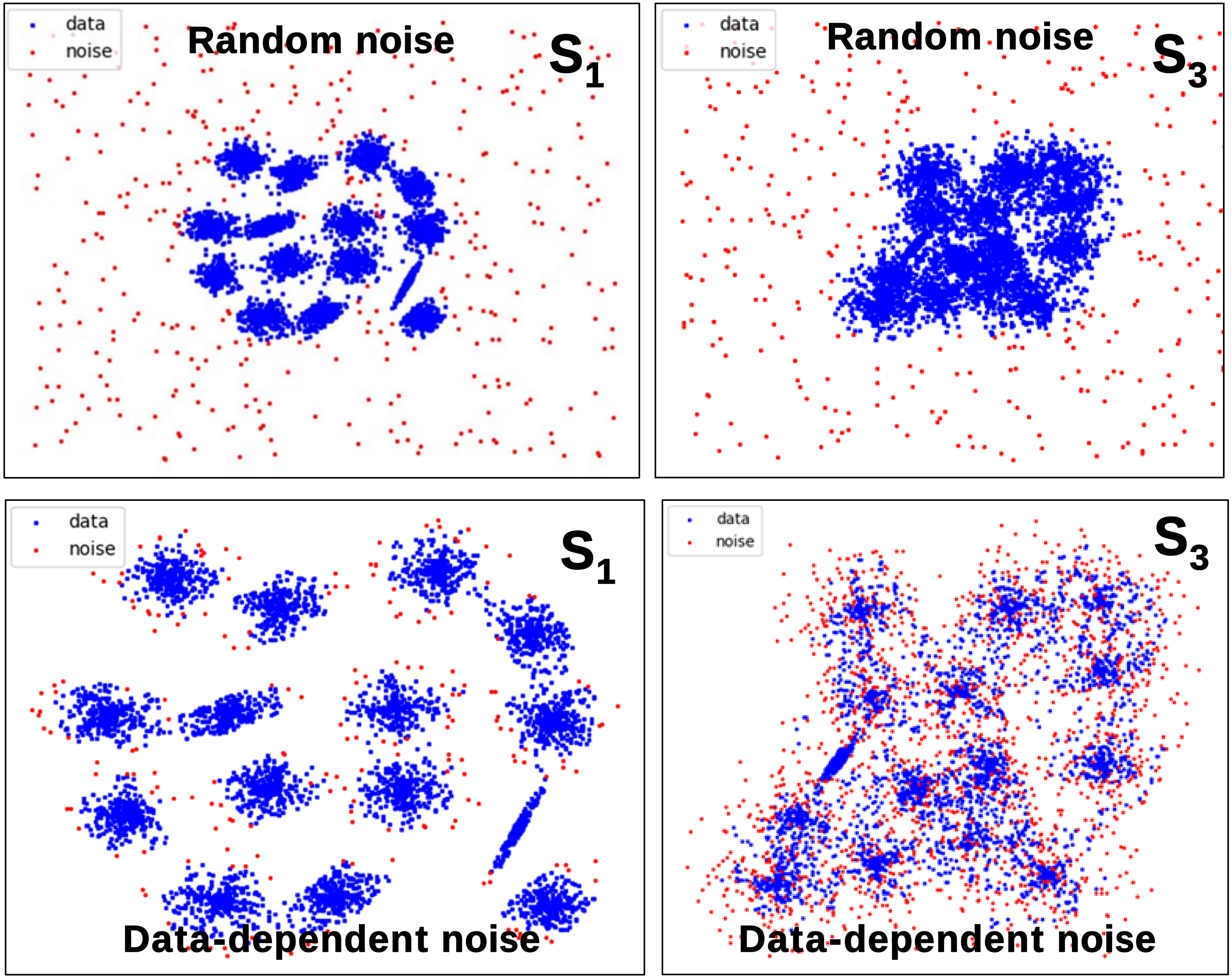
# Experiments

# Datasets

P. Fränti and S. Sieranoja, "K-means properties on six clustering benchmark datasets", *Applied Intelligence,* 2018.



# Noise types


data
noise
Random noise
S1
data
noise
Random noise
S3
data
noise
S1
Data-dependent noise
data
noise
S3
Data-dependent noise

# Results for noise type 1

Centroid Index (CI)

| Pre-process: | Combination | S1 | S2 | S3 | S4 | A1 | A2 | A3 | Un | Av. |
|---|---|---|---|---|---|---|---|---|---|---|
| none | RS | 4 | 4 | 4 | 3 | 6 | 7 | 14 | 4 | 5.8 |
| | KM | 4 | 3 | 3 | 3 | 5 | 8 | 13 | **2** | 5.1 |
| noise removal | LOF+RS | 3 | 4 | 3 | 2 | 5 | 5 | 9 | **2** | 4.1 |
| | LOF+KM | 2 | 4 | 3 | 3 | 4 | 6 | 10 | 3 | 4.4 |
| | ODIN+RS | 4 | 4 | 4 | 3 | 5 | 7 | 14 | 3 | 5.5 |
| | ODIN +KM | 4 | 4 | 4 | 4 | 6 | 8 | 11 | **2** | 3.5 |
| medoid-shift | medoid+RS | **0** | **0** | 2 | 2 | **0** | **2** | 4 | **2** | 1.5 |
| | medoid+KM | 1 | 1 | **1** | **1** | **0** | **2** | **3** | **2** | **1.4** |
| | mean+RS | 4 | 3 | 6 | 3 | 3 | 7 | 13 | 3 | 5.3 |
| | mean+KM | 4 | 4 | 4 | 2 | 4 | 7 | 14 | **2** | 5.1 |

**CI reduces from 5.1 to 1.4**

KM: K-means with random initialization
RS: P. Fränti, "Efficiency of random swap clustering", *Journal of Big Data, 5:13, 1-29, 2018.*

# Results for noise type 2

Centroid Index (CI)

| Pre-process: | Combination | S1 | S2 | S3 | S4 | A1 | A2 | A3 | Un | Av. |
|---|---|---|---|---|---|---|---|---|---|---|
| **none** | RS | 4 | 4 | 4 | 3 | 5 | 7 | 13 | 3 | 5.4 |
| | KM | 4 | 4 | 4 | 3 | 4 | 6 | 12 | 3 | 5.0 |
| **noise removal** | LOF+RS | 4 | 4 | 4 | 3 | 5 | 7 | 13 | **2** | 5.3 |
| | LOF+KM | 4 | 4 | 4 | 3 | 5 | 7 | 11 | **2** | 5.0 |
| | ODIN+RS | 4 | 4 | 4 | 4 | 5 | 8 | 13 | 3 | 5.6 |
| | ODIN+KM | 4 | 4 | 4 | 3 | 4 | 7 | 11 | **2** | 4.8 |
| **medoid-shift** | medoid+RS | **0** | **0** | **1** | 2 | **0** | **1** | 5 | 3 | 1.5 |
| | medoid+KM | **0** | **0** | **1** | **1** | **0** | **1** | **4** | **2** | **1.1** |
| | mean+RS | 4 | 4 | 4 | 3 | 4 | 7 | 11 | 3 | 5.0 |
| | mean+KM | 2 | 4 | 4 | 3 | 3 | 5 | 12 | 3 | 4.5 |

**CI reduces from 5.0 to 1.1**

KM: K-means with random initialization

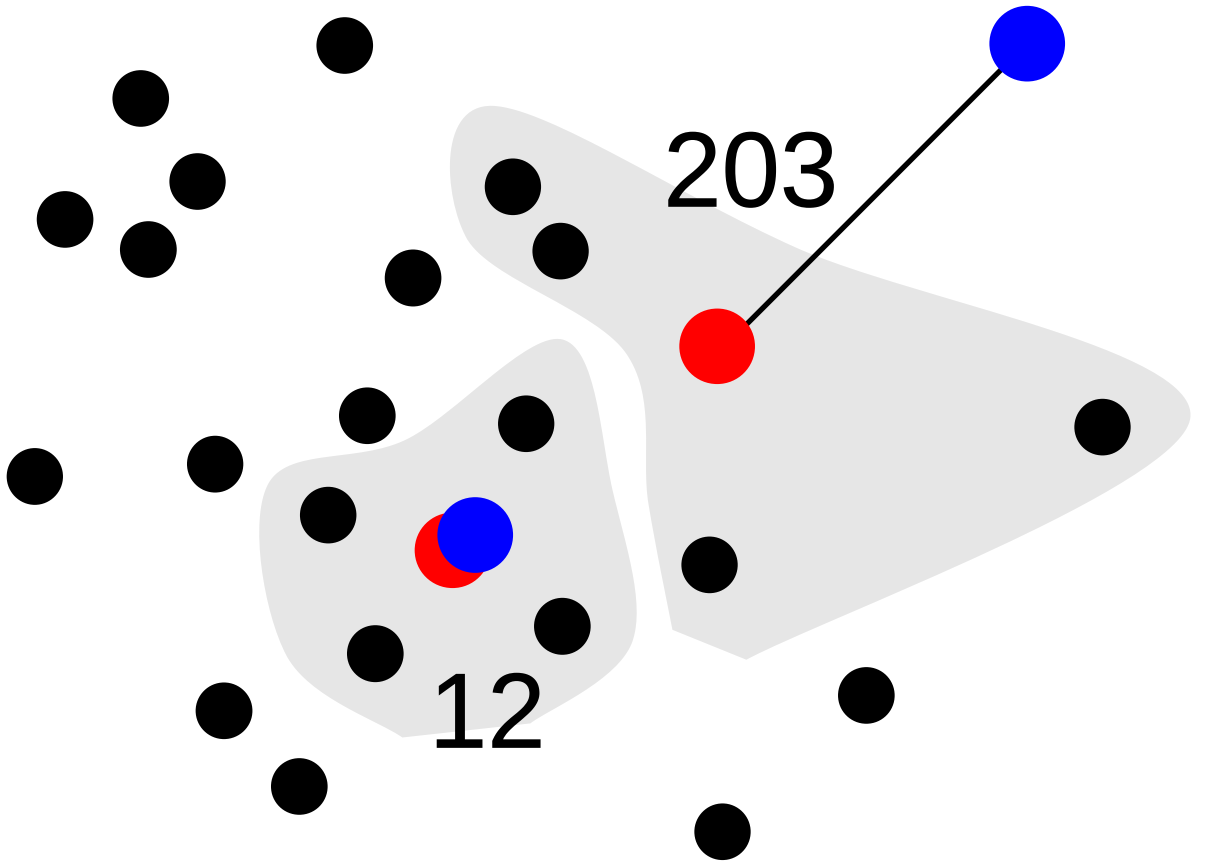
RS: P. Fränti, "Efficiency of random swap clustering", *Journal of Big Data, 5:13, 1-29, 2018.*

# Part II:
# Outlier detection

Yang, Rahardja, Fränti, "Mean-shift outlier detection",
*Int. Conf. Fuzzy Systems and Data Mining (FSDM)*, November 2018.

# Outlier scores

$D = |x - y|$



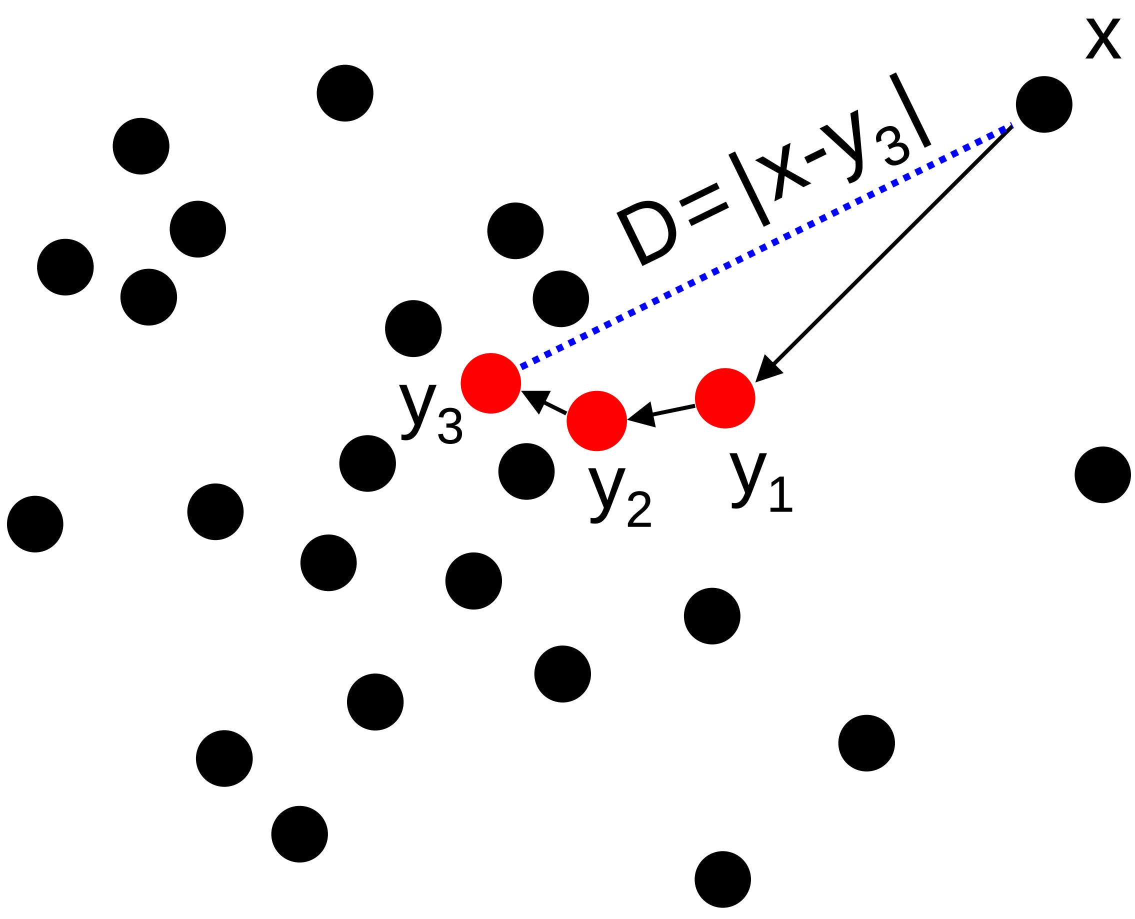

**k=4**

# Mean-shift for Outlier Detection

**Algorithm: MOD**

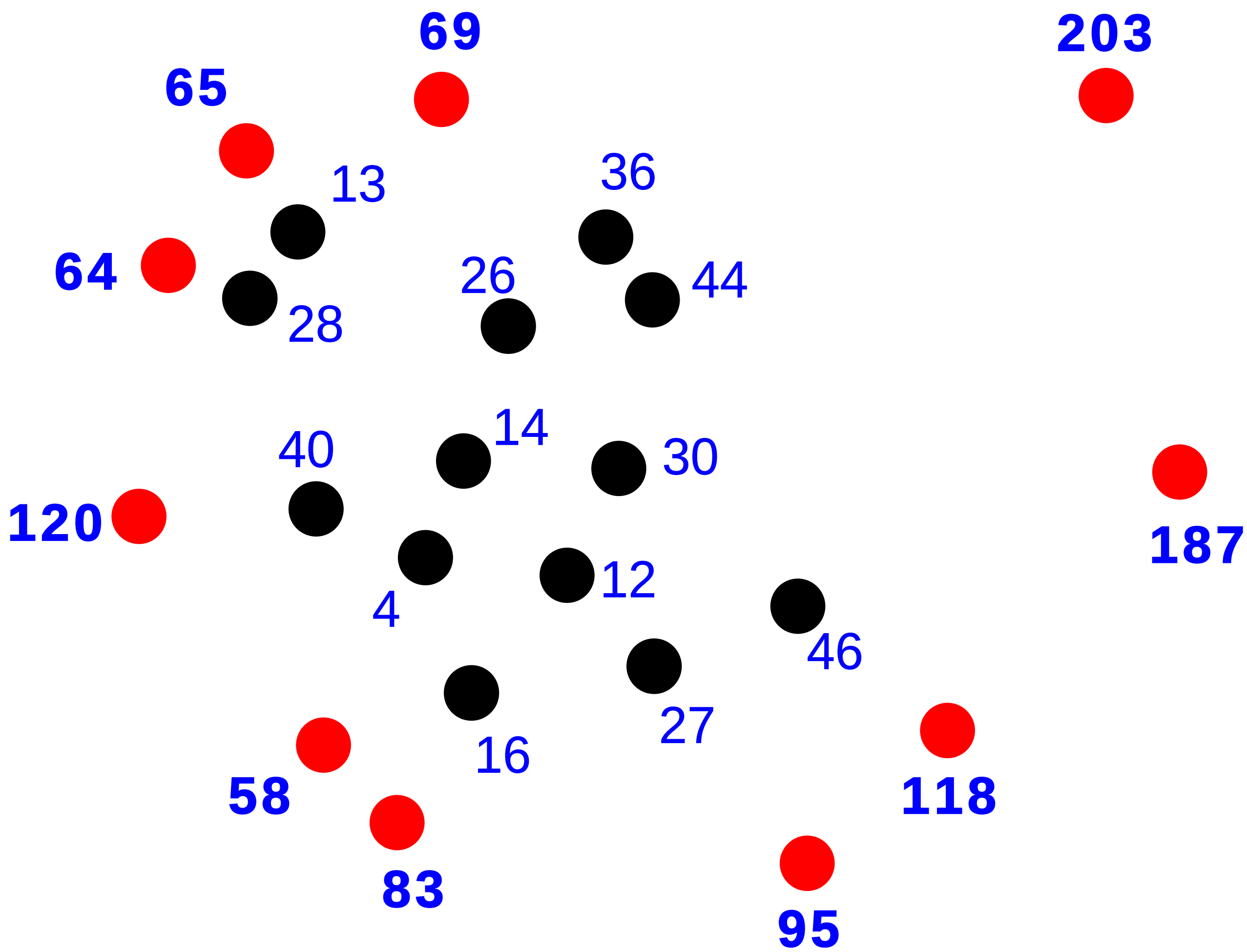
1. Calculate Mean-shift $\mathbf{y}$ for every point $\mathbf{x}$
2. Outlier score $\mathbf{D_i} = |\mathbf{y_i} - \mathbf{x_i}|$
3. SD = Standard deviation of all $\mathbf{D_i}$
4. IF $\mathbf{D}_i$ > SD THEN $\mathbf{x_i}$ is OUTLIER

# Result after three iterations



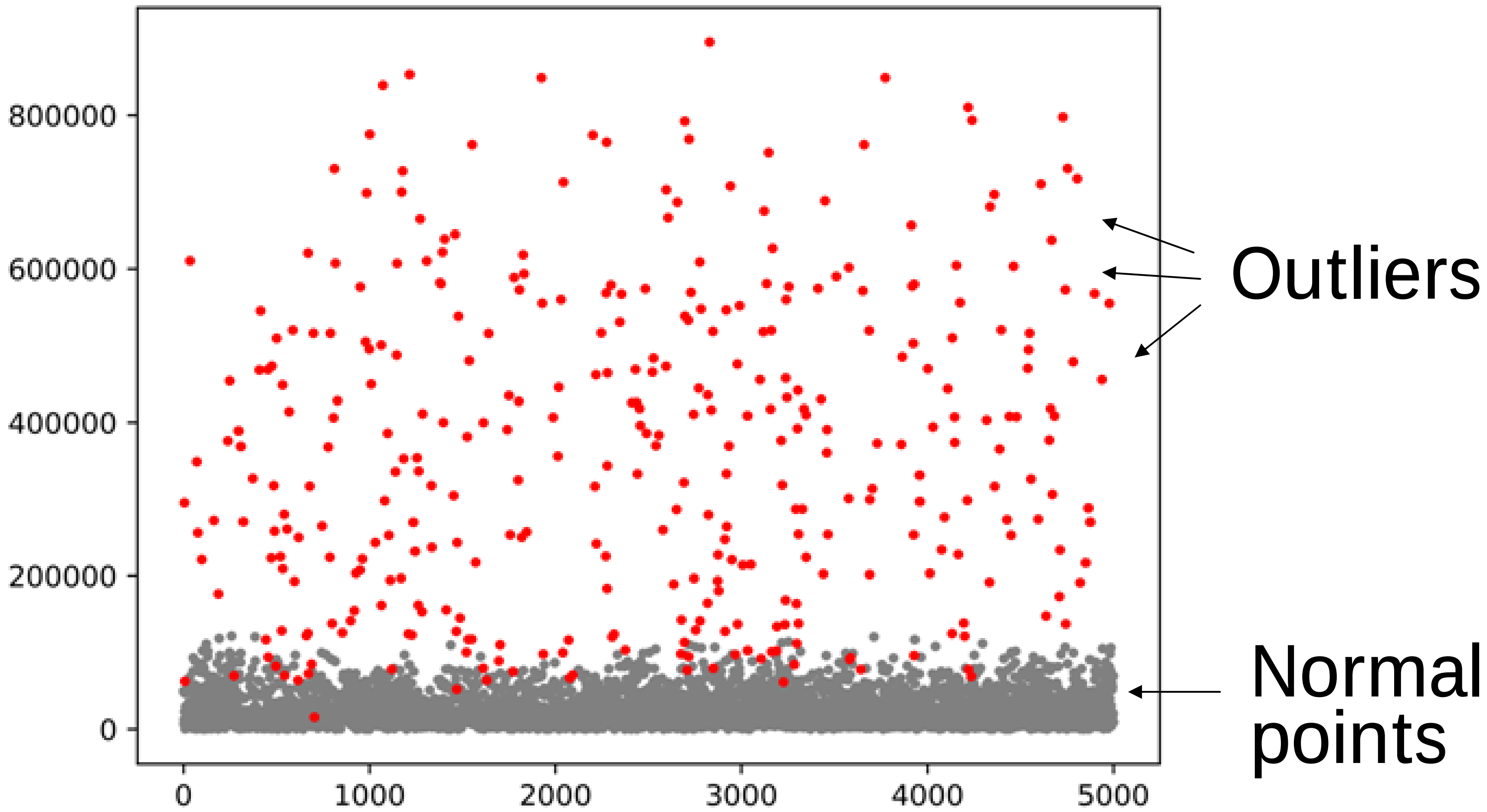
# Outlier scores

$D=|x-y|$




SD=52

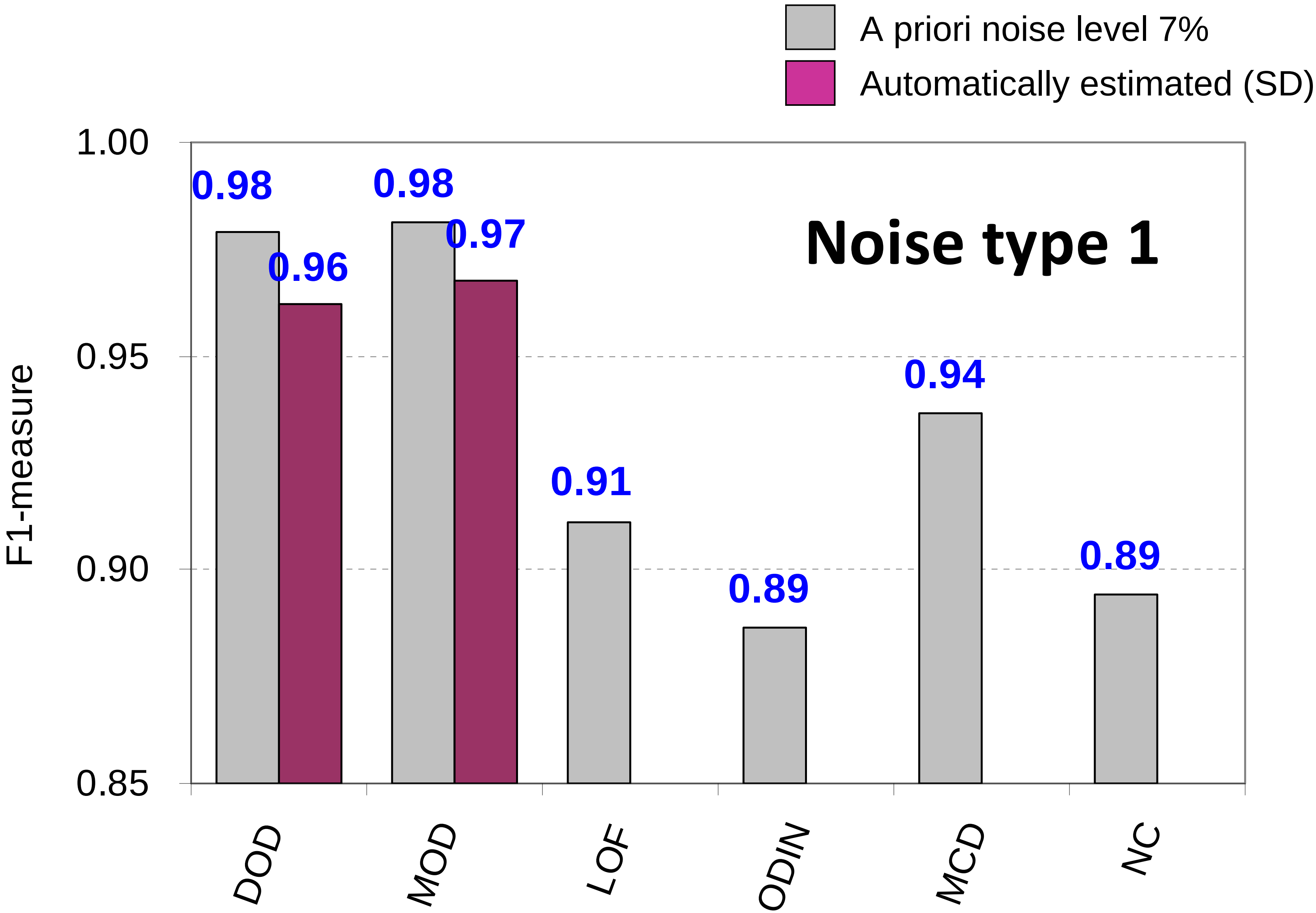
# Distribution of the scores



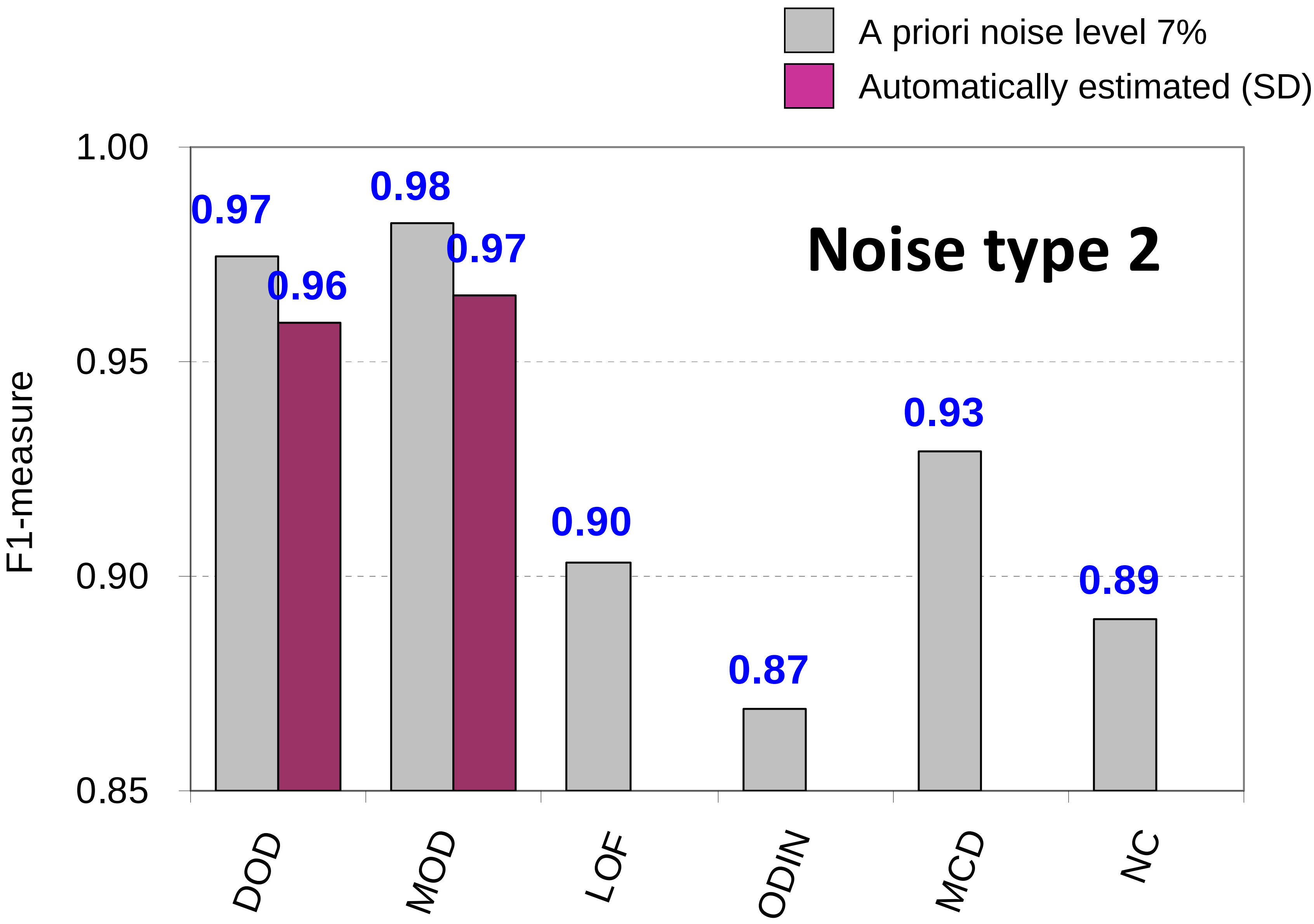
# Experimental comparisons

| Algorithms | Ref | Type | Parameters | Year | Publication |
|---|---|---|---|---|---|
| LOF | [9] | Density-based | $k$, top-$N$ | 2000 | ACM SIGMOD |
| KNNG | [7] | Distance-based | $k$, top-$N$ | 2004 | Int. Conf. on Pattern Recognition |
| MCD | [11] | Statistical testing | top-$N$ | 1984 | J. Am. Stat. Assoc. |
| NC | [12] | Math. optimization | $k$, top-$N$ | 2018 | IEEE-TNNLS |
| MOD | new | Shifting-based | $k$ | 2018 | Int. Conf. Fuzzy Syst. Data Mining |

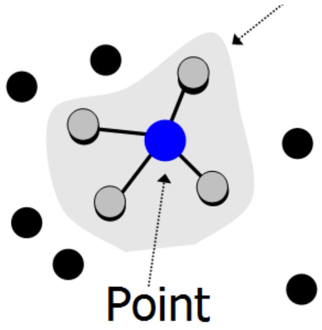
# Results for Noise type 1



# Results for Noise type 2



# Conclusions




## Why to use:

- Simple but effective!

## How it performs:

- Outperforms existing methods!

98-99%

## Usefulness:

- No threshold parameter needed!