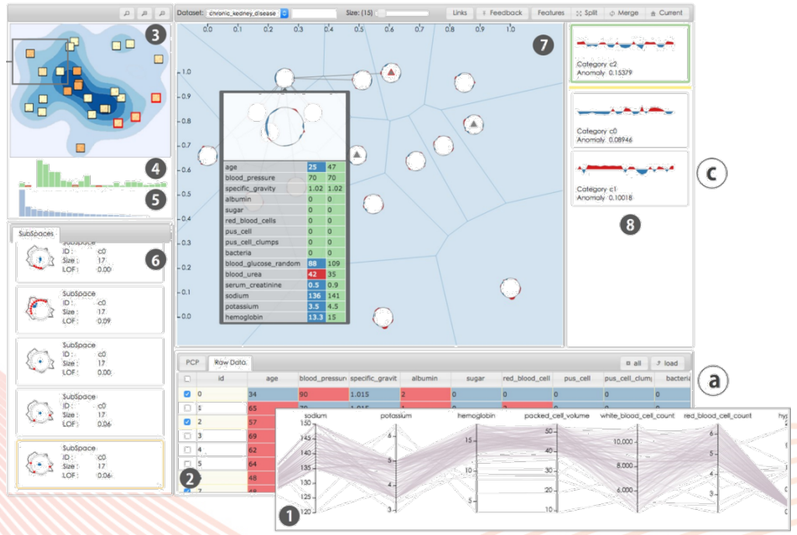
VIS 2017



RCLens: Interactive Rare Category Exploration and Identification

Hanfei Lin[1], **Siyuan Gao[1]**, David Gotz[2], Fan Du[3], Jingrui He[4], and Nan Cao[1]

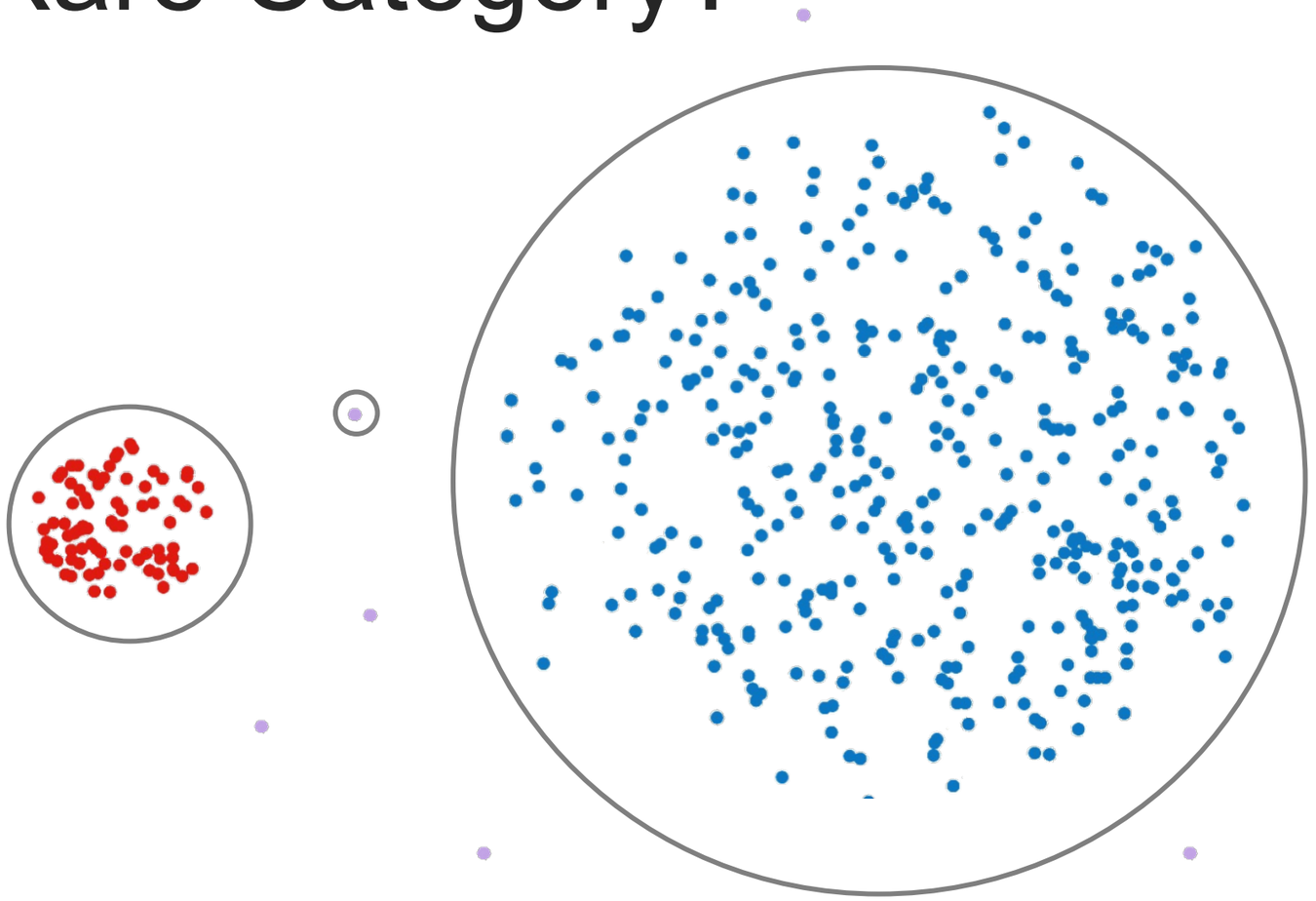
Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

What is Rare Category?



A small group of points that have the property of compactness and isolation

Application

Financial Fraud Detection

Network Security

Personal Medicine

What is Rare Category Identification?

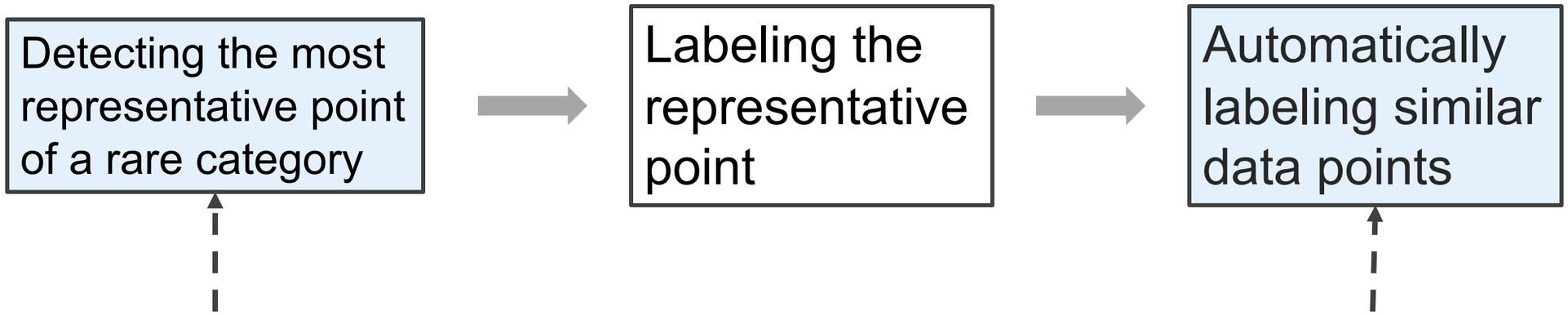
- Different from outlier identification, which finds the isolated data items. Rare category identification tends to find one or more minority classes which are often **self-similar**, potentially forming compact clusters.

- It is an **active learning process** which requires human to label sampled instances in each detected rare category

Typical Steps of Rare Category Identification

Typical Steps of Rare Category Identification

NNDM



Representative point is the data item that has the similar neighborhood density to a given density constant *p* defined by *a priori* knowledge

Label data that are center around the representative point within the radius of *np,* where *n* is the total number of data points

He, J., et al. NIPS (2007)

NNDM

Detecting the most

Labeling the

Automatically

Requires *a priori* knowledge, which is unknown in real world situations

Imprecise labeling of similar points will increase the number of labeling iterations

Repres … around
that h … within
densit … *n* is the
p defined by the prior knowledge … total number of data points

He, J., et al. NIPS (2007)

Challenge

- How to find the representative points without any *a priori* knowledge ?
- How to adaptively determine boundary of rare category ?
- How to minimize the total number of labeling iterations ?

Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

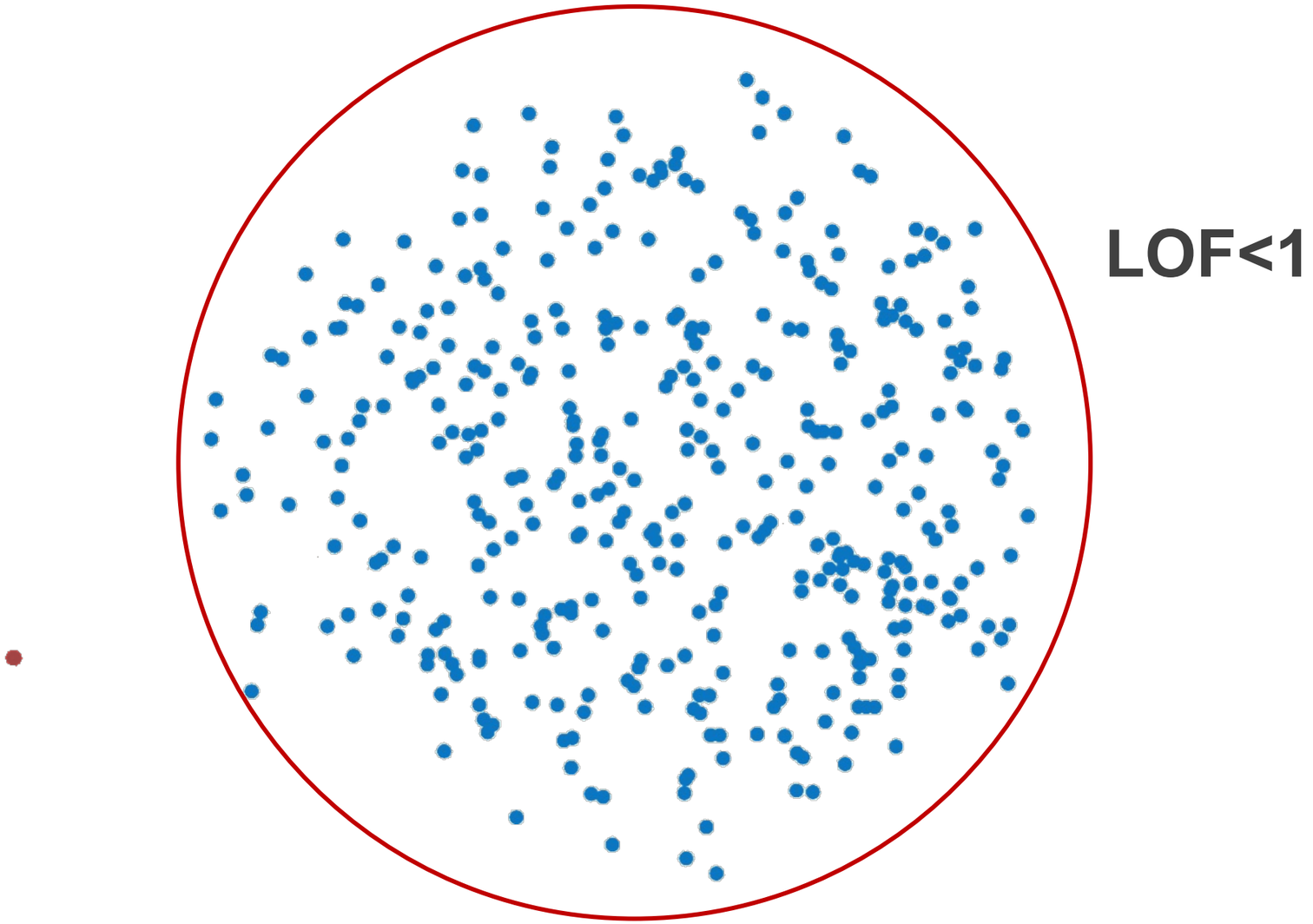
Local Outlier Factor Algorithm (LOF)

$$LOF_k(a) = \frac{\sum_{b \in NN_k(a)} \frac{lrd_{k(b)}}{lrd_k(a)}}{k}$$

$$lrd_k(t) = \left[\frac{\sum_{s \in NN_k(t)} dist_k(t,s)}{k}\right]^{-1},$$

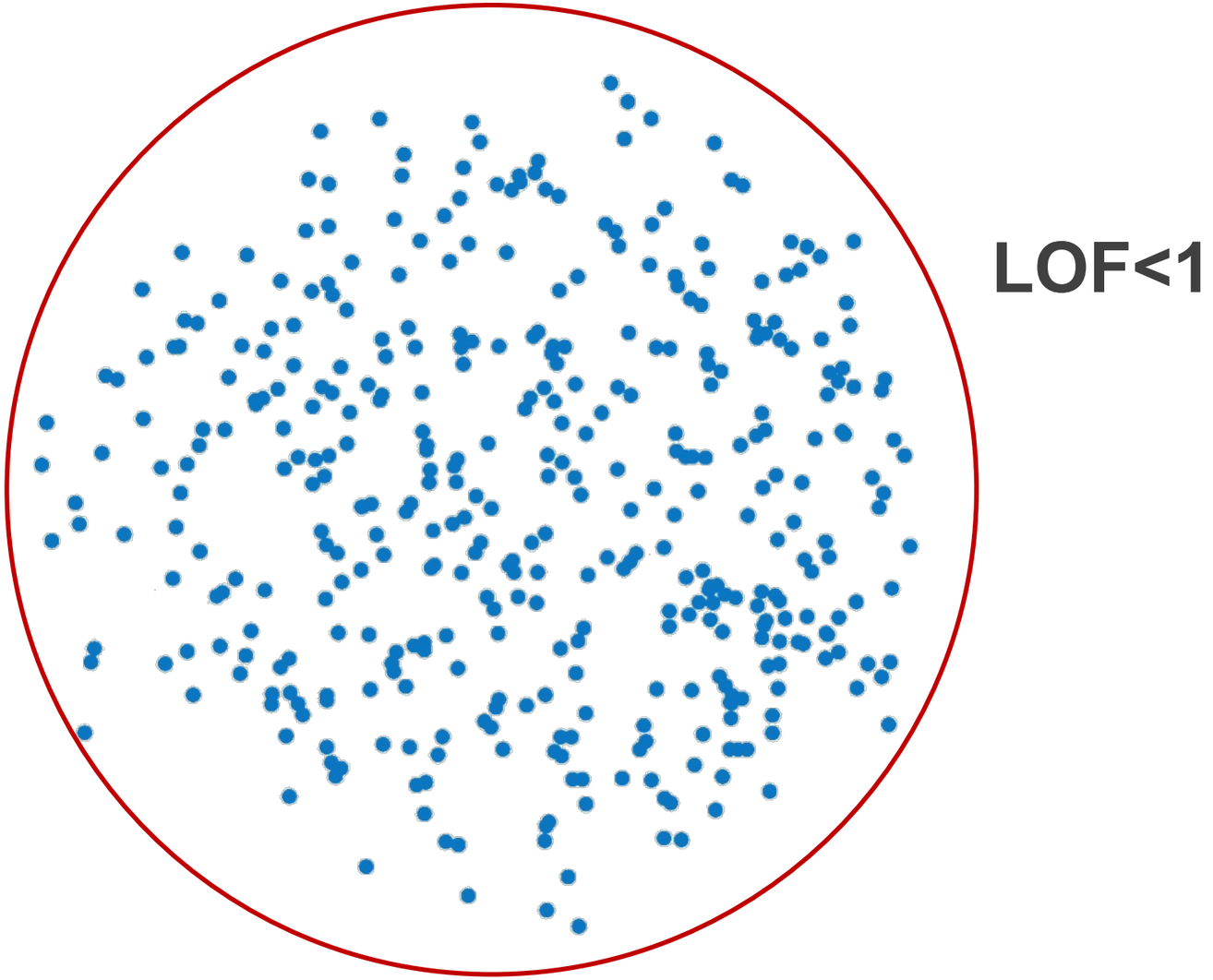
$$dist_k(a,b) = max(d_k(b), d(a,b))$$

Local Outlier Factor Algorithm (LOF)

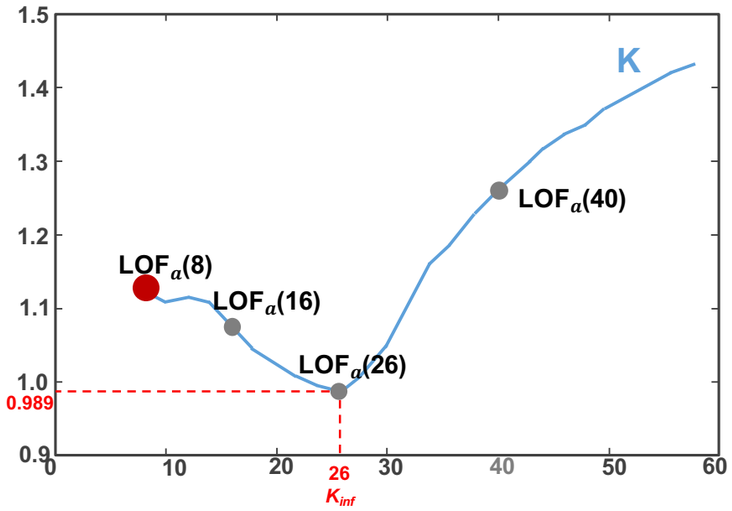


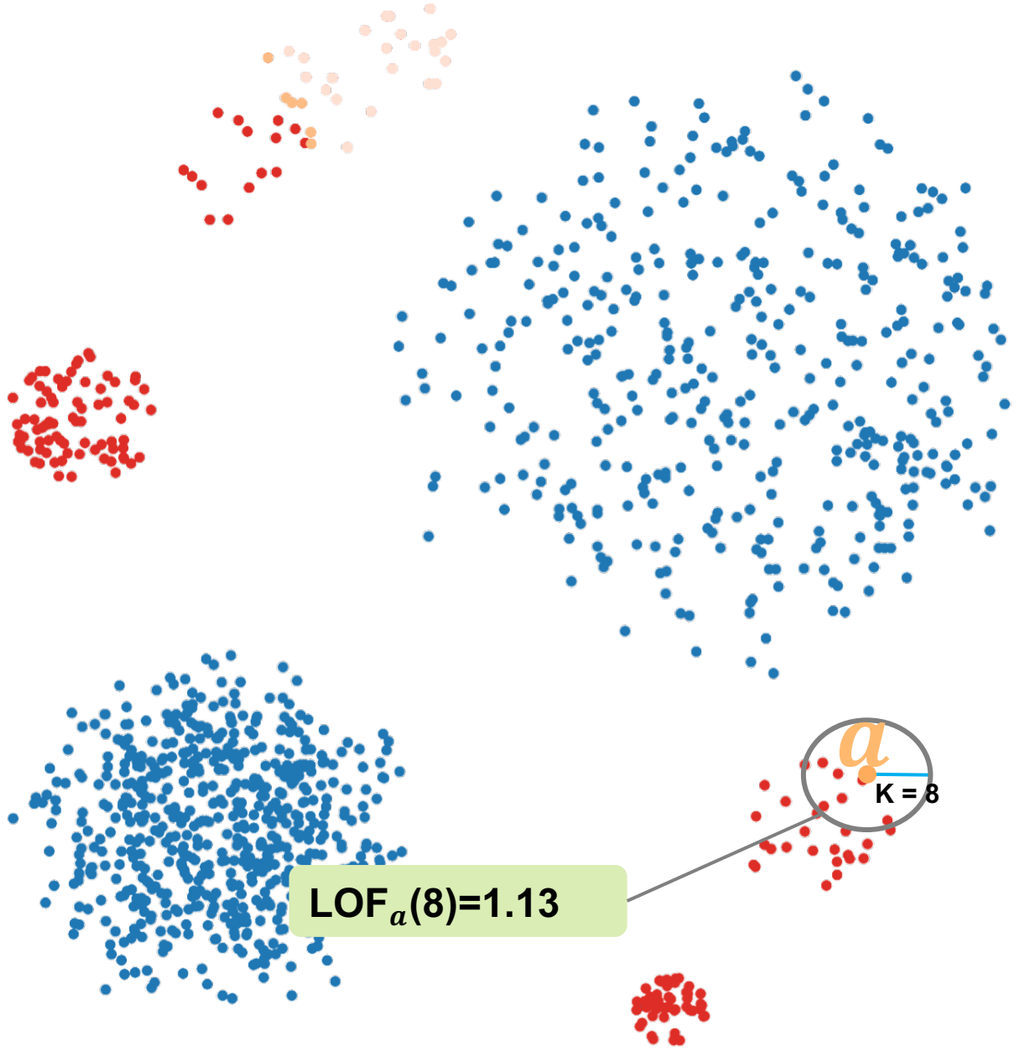
LOF<1

Local Outlier Factor Algorithm (LOF)

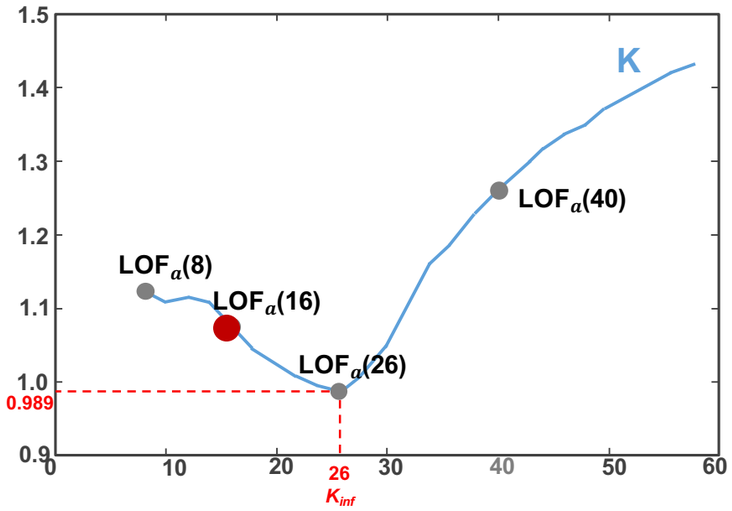


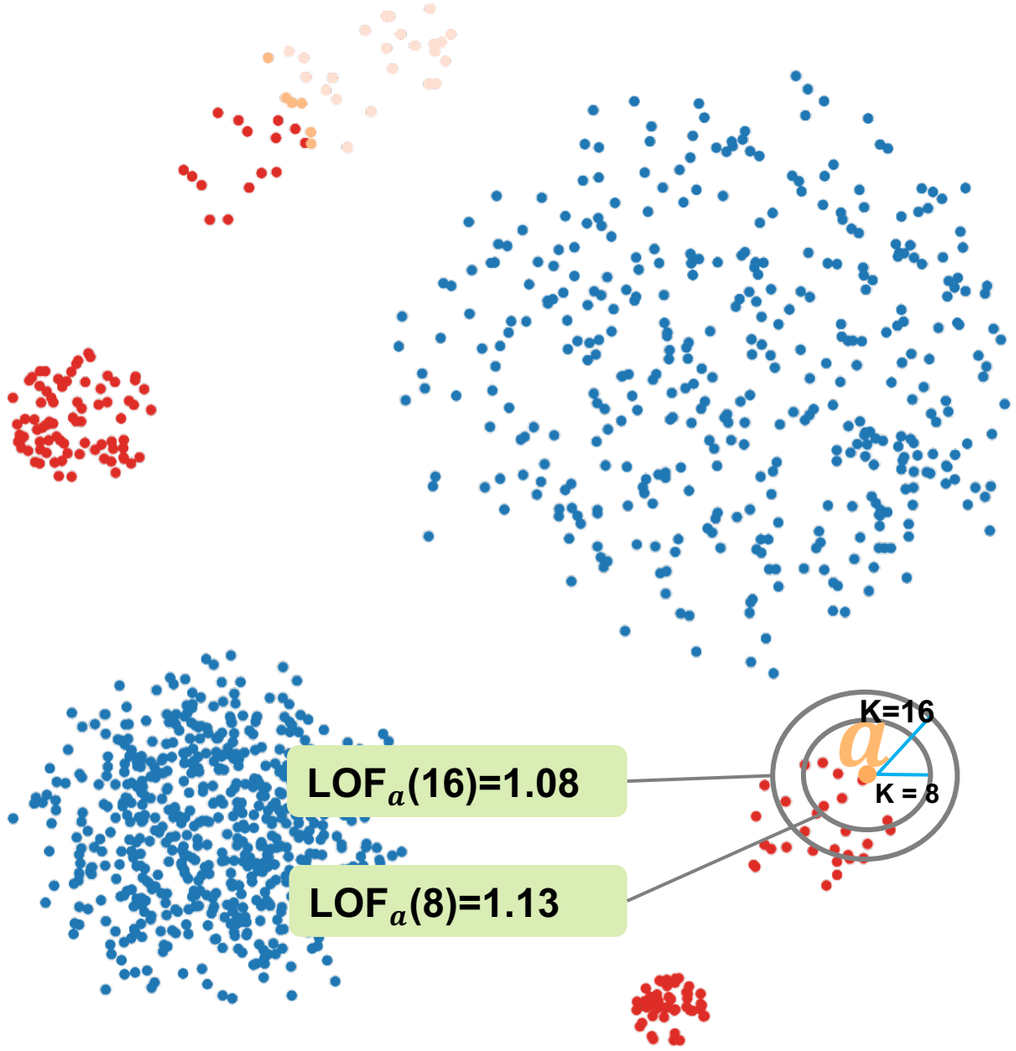
Local Outlier Factor Algorithm (LOF)



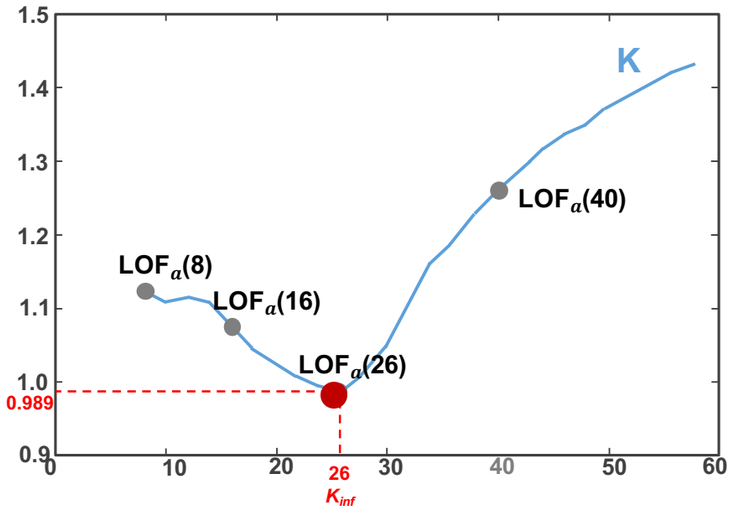


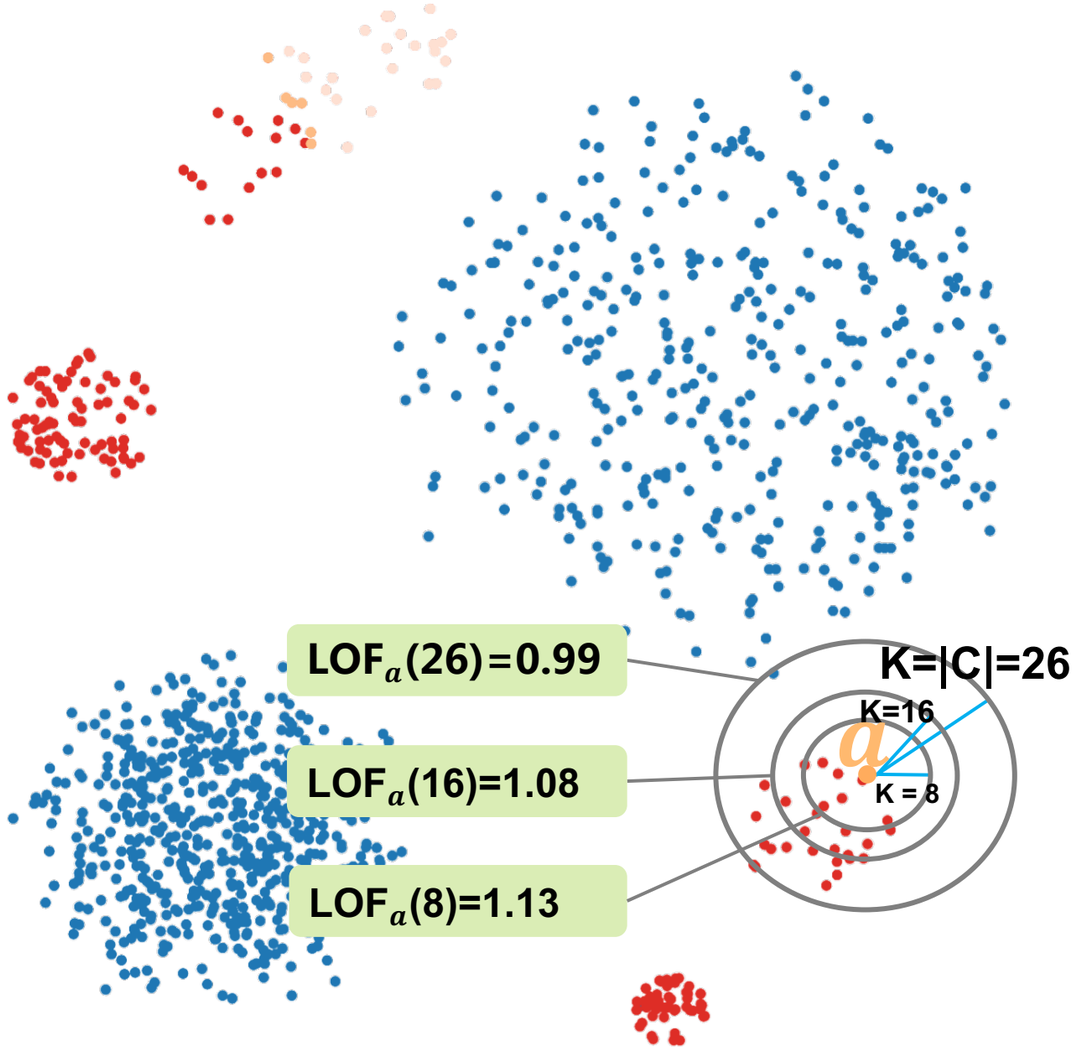
Local Outlier Factor Algorithm (LOF)



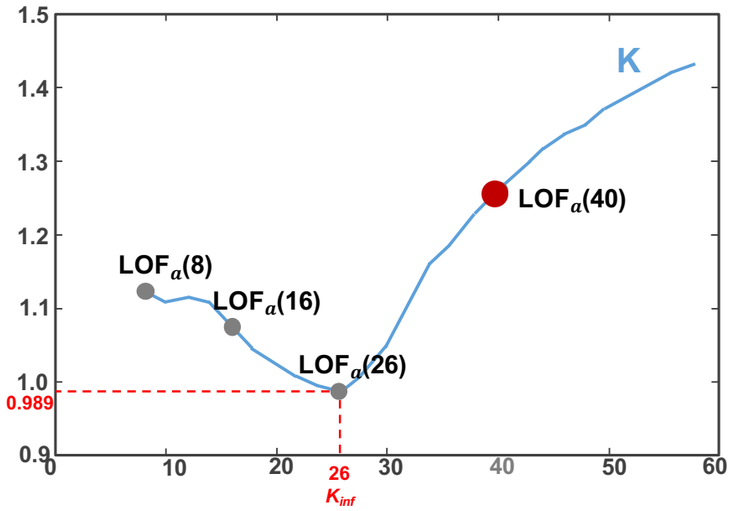


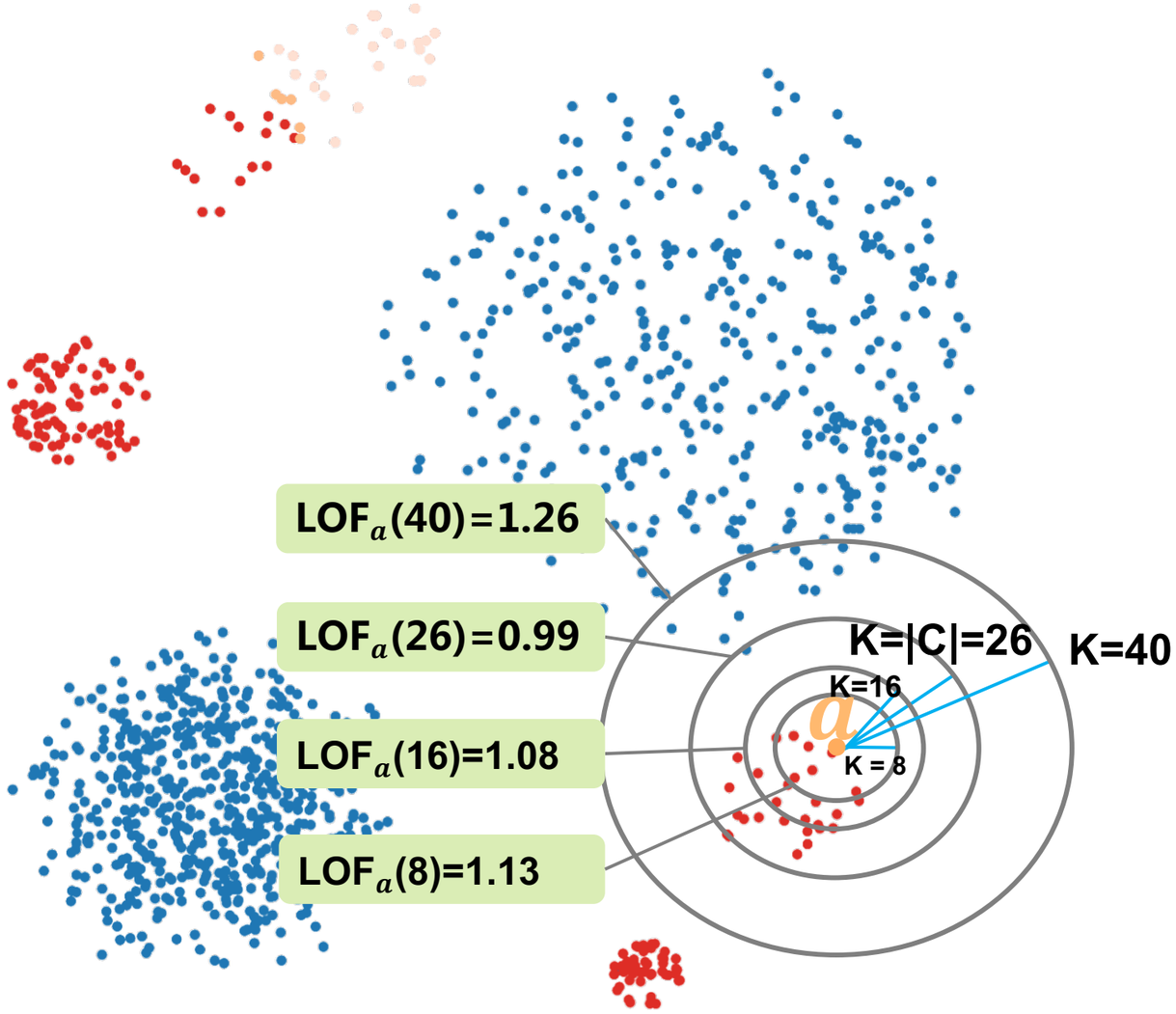
Local Outlier Factor Algorithm (LOF)





Local Outlier Factor Algorithm (LOF)

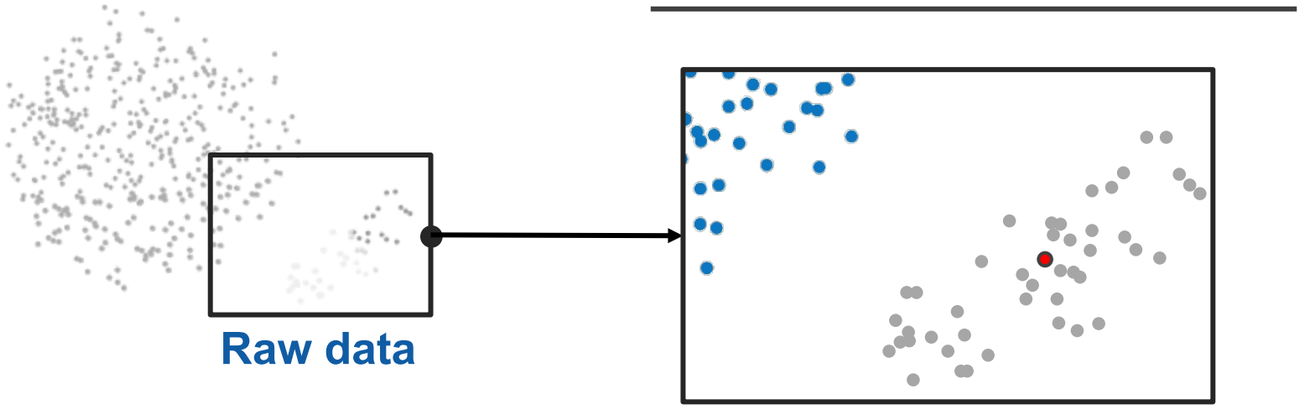




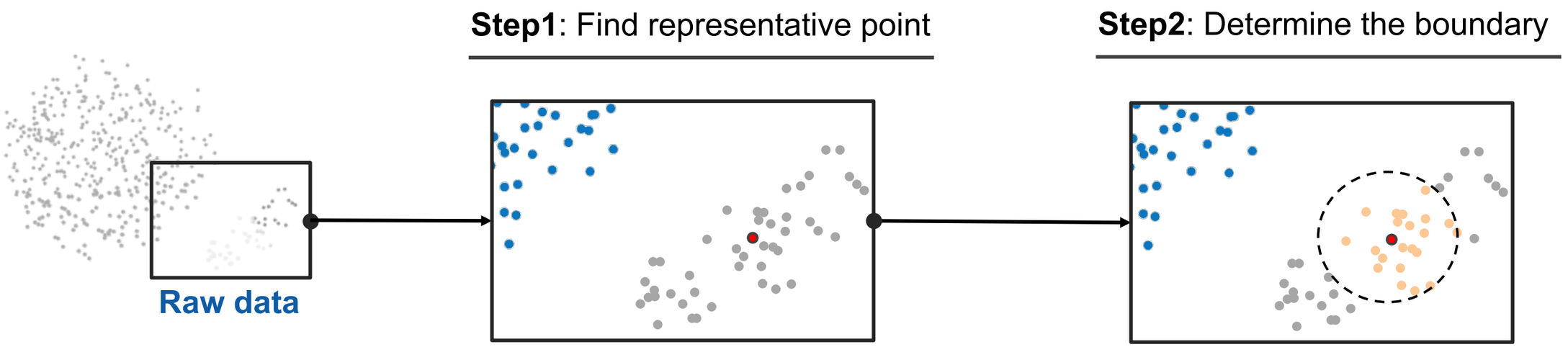
LOF-based Rare Category Detection Algorithm (LOFRCD)

LOFRCD Algorithm

Step1: Find representative point



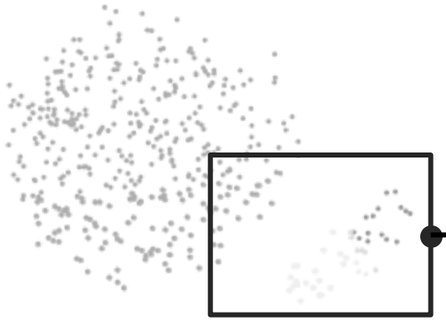
LOFRCD Algorithm



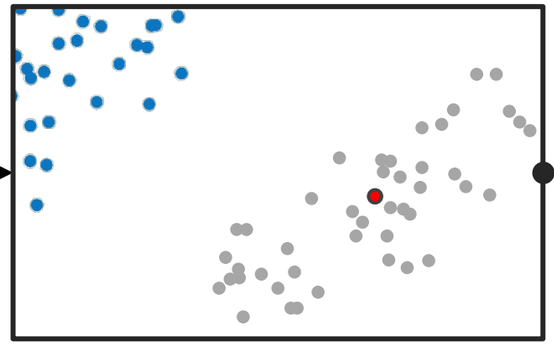
LOFRCD Algorithm

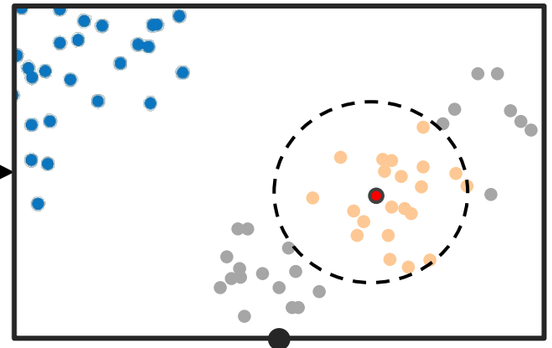
Step1: Find representative point

Step2: Determine the boundary

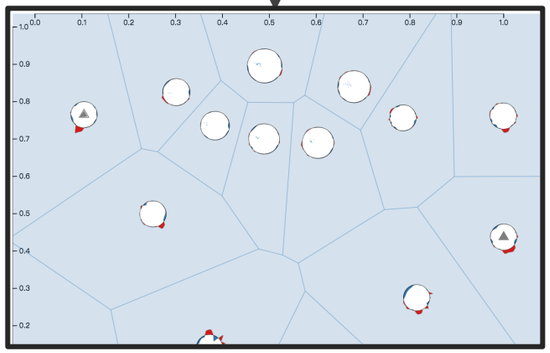


Raw data

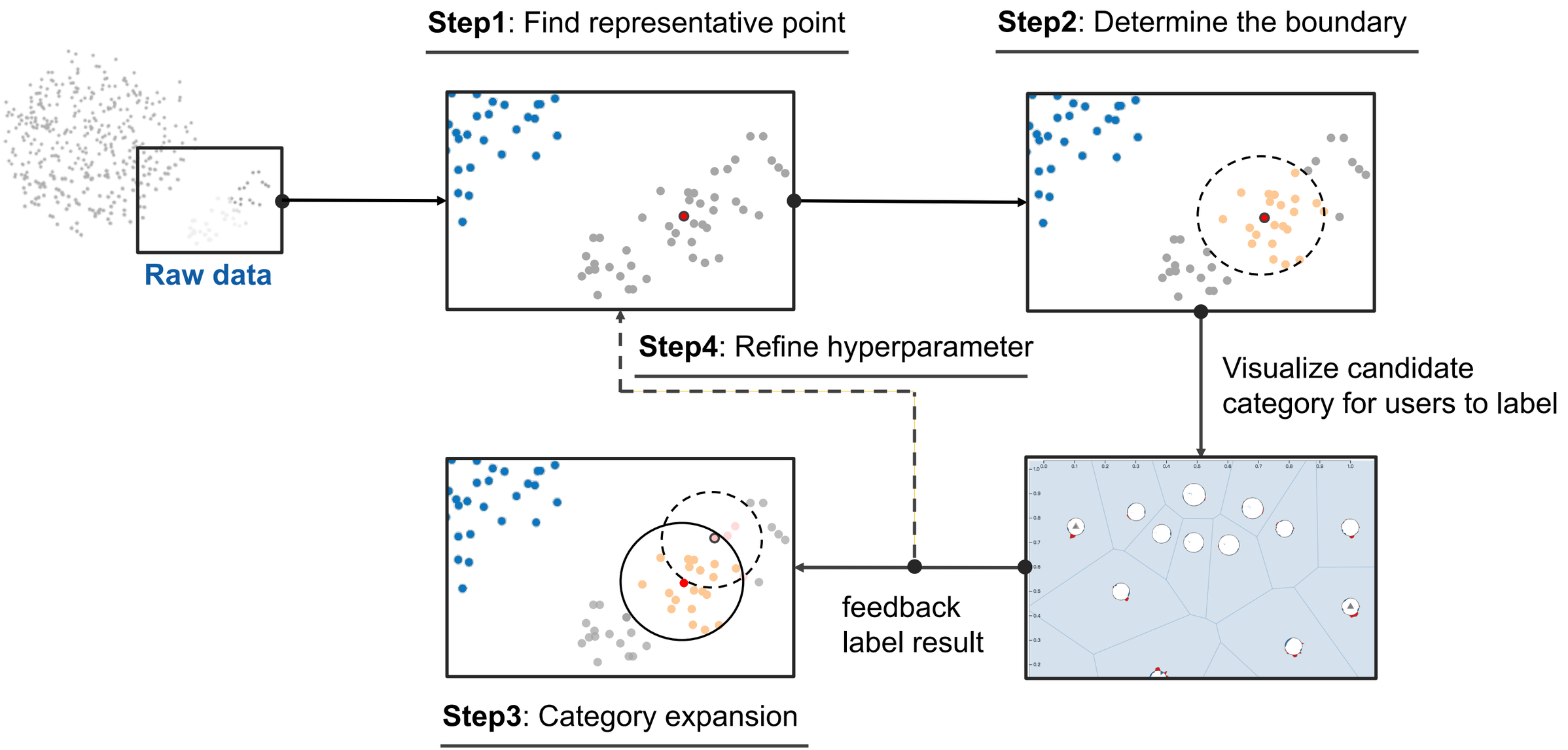




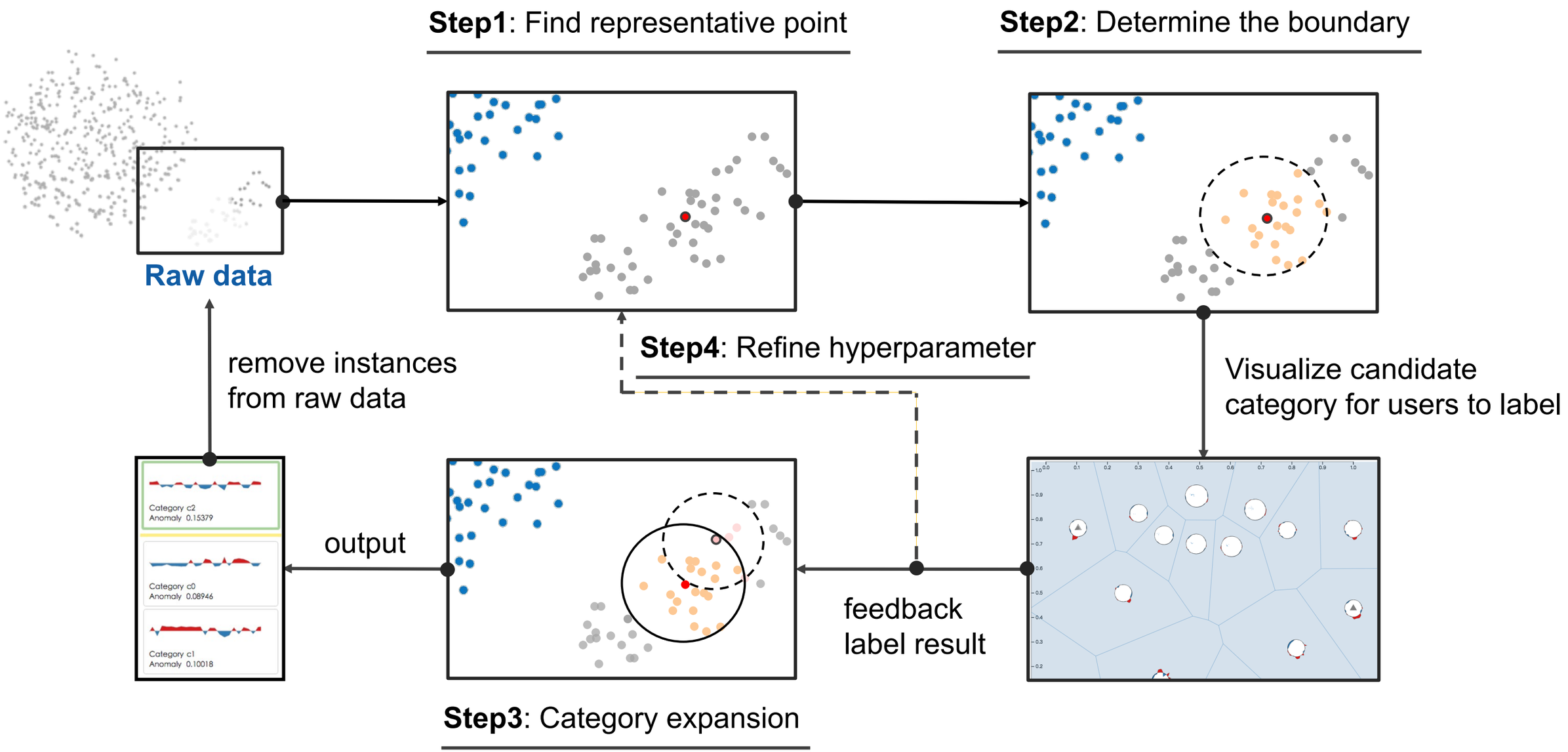
Visualize candidate category for users to label



LOFRCD Algorithm



LOFRCD Algorithm



Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should represent an isolated minority class

C_1 : the isolation of a minority class

$$C_{iso}(a) = \frac{LOF_{k_{inf}+1}(a)}{LOF_{k_{inf}}(a)}$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should represent an isolated minority class
- Representative point's neighbors should also represent the class, i.e., also have a large C_1 score.

C_1 : the isolation of a minority class

$$C_{iso}(a) = \frac{LOF_{k_{inf}+1}(a)}{LOF_{k_{inf}}(a)}$$

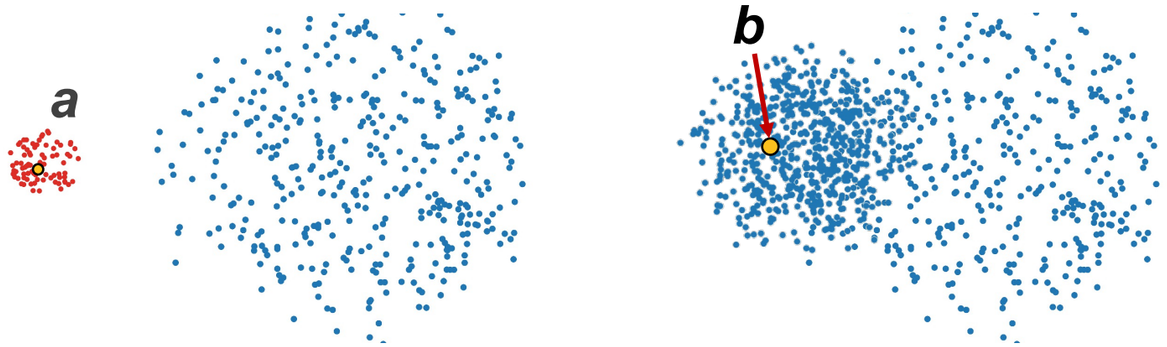
$$C_1(a) = \frac{\sum_{b \in NN_k(a)} C_{iso}(b)}{k_{inf}} * C_{iso}(a)$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should represent an isolated minority class
- Representative point's neighbors should also represent the class, i.e., also have a large C_1 score.

C_1 : the isolation of a minority class



$$C_1(a) > C_1(b)$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should be close to all its neighbors within the category, and distant from neighbors outside the category

C_2 : Be close to all its neighbors within the category, and distant from neighbors outside the category

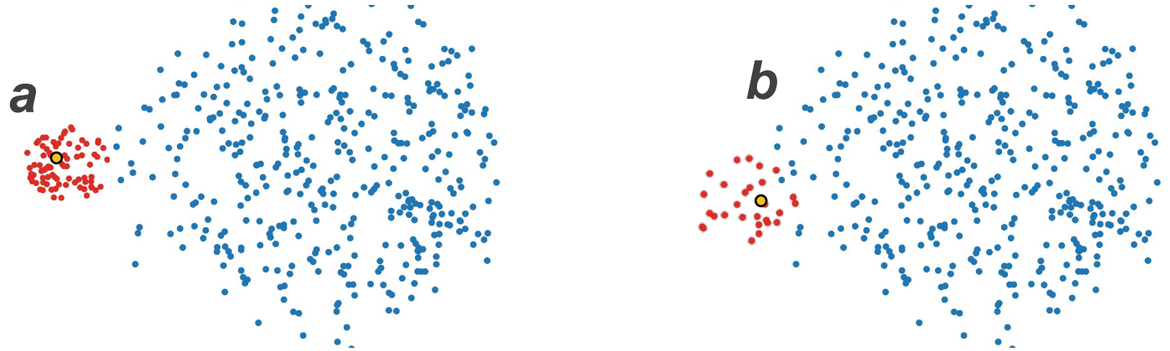
$$C_2(a) = \frac{d_{k_{inf}+1}(a)}{d_{k_{avg}}(a)}$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should be close to all its neighbors within the category, and distant from neighbors outside the category

C_2 : Be close to all its neighbors within the category, and distant from neighbors outside the category



$$C_2(a) > C_2(b)$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Representative point should have a k_{inf} value that is similar to its k_{inf}-neighbors' k_{inf} values

C_3 : a should have a similar k_{inf} value within the category

$$C_3(a) = exp\left[- \left\| \frac{k_{inf}^{a}}{\frac{\sum_{b \in NN_{k_{inf}^{a}}(a)} k_{inf}^{b}}{k_{inf}^{a}}} - 1 \right\| \right]$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

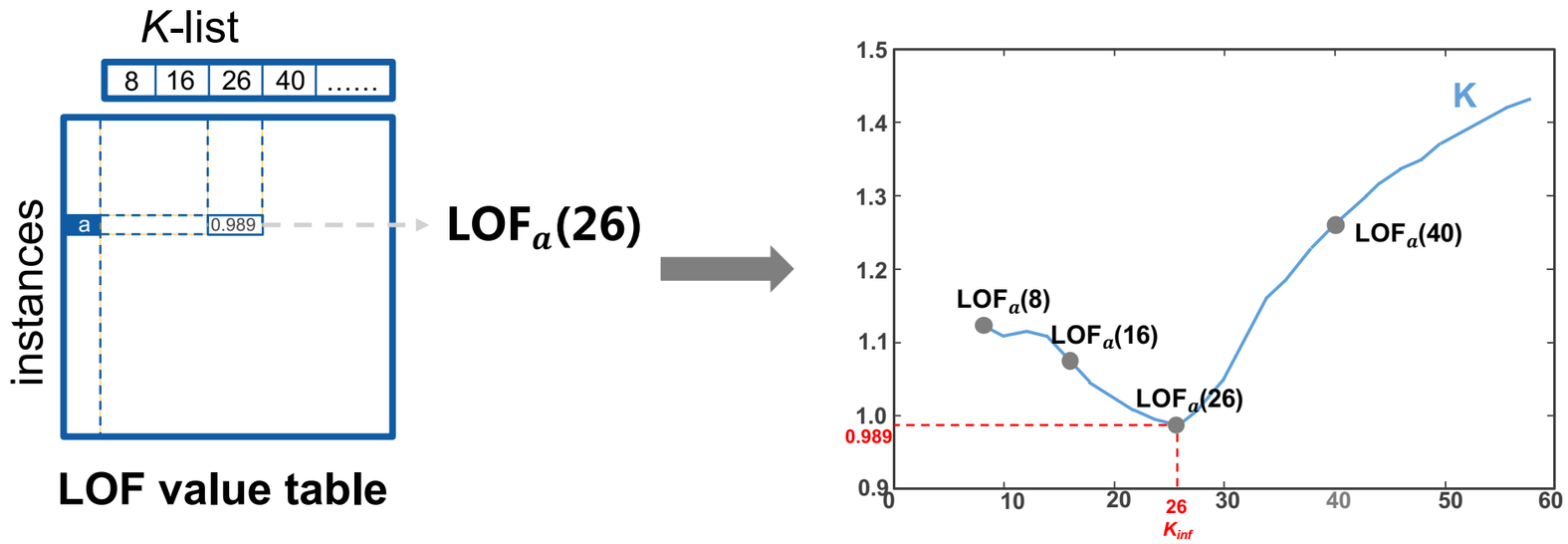
- Calculating the overall confidence scores
- Representative point should have a high overall confidence score

$$C_{total} = \sqrt[3]{C_1 * C_2 * C_3}$$

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

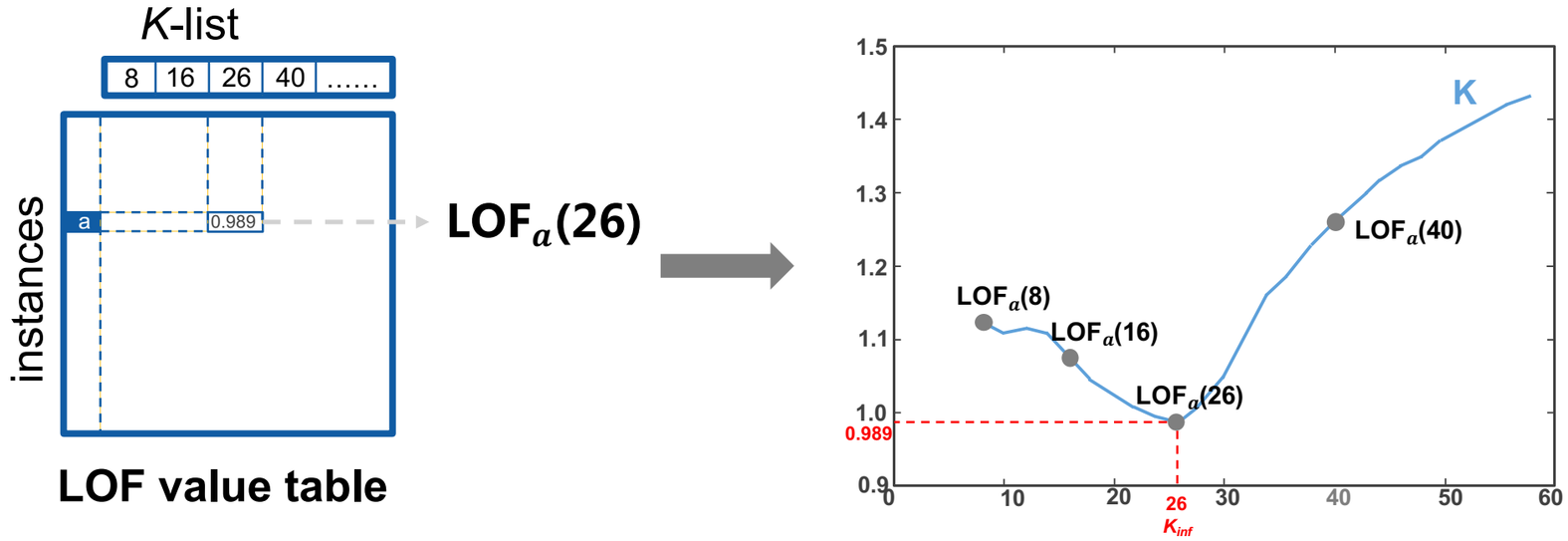
- *K*-list is the list contains all the k values for LOF algorithm that we will test to determine the inflection point



Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

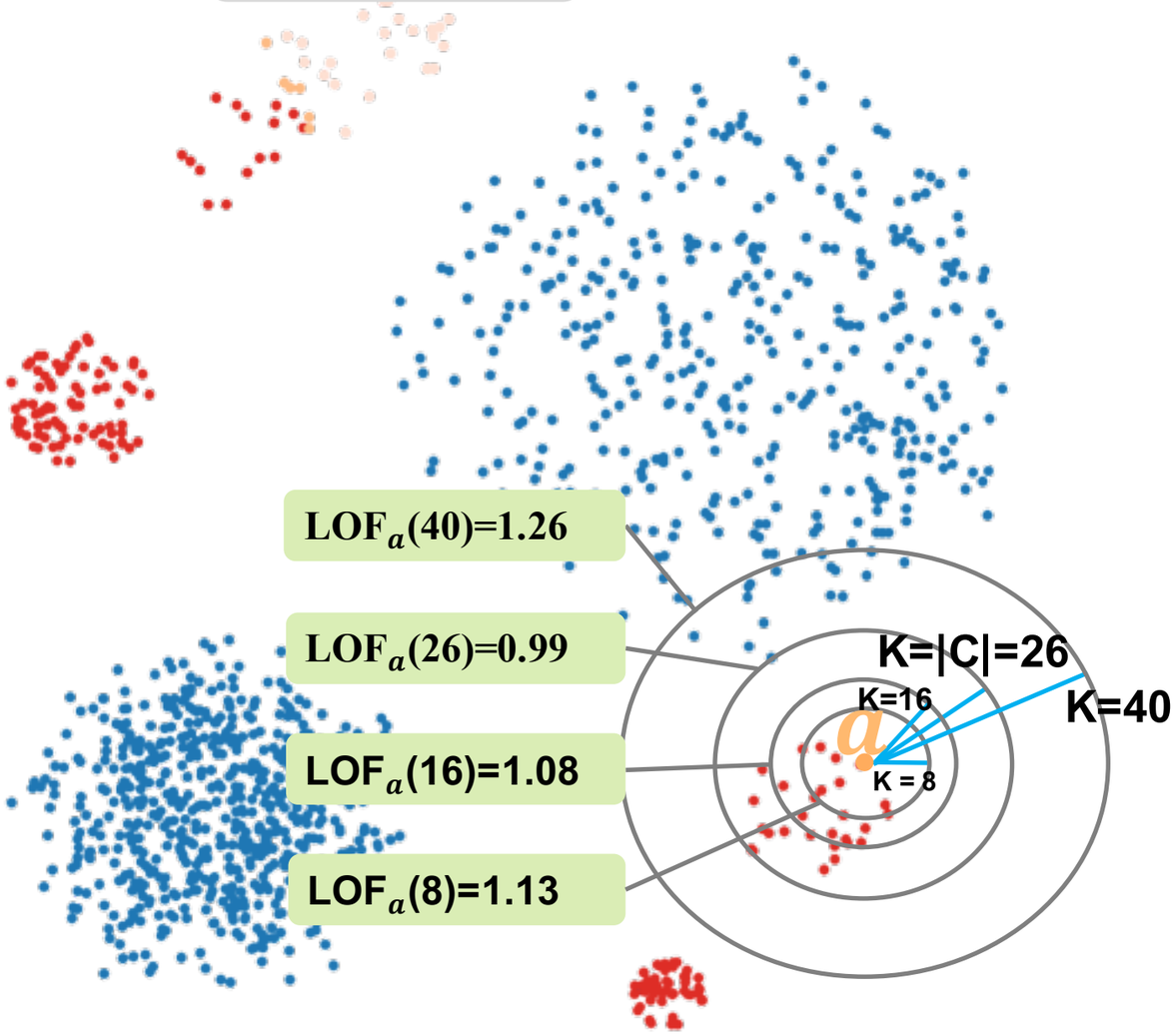
- K-list is the list contains all the k values for LOF algorithm that we will test to determine the inflection point
- k_{inf} is the inflection point in the LOF score curve



Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

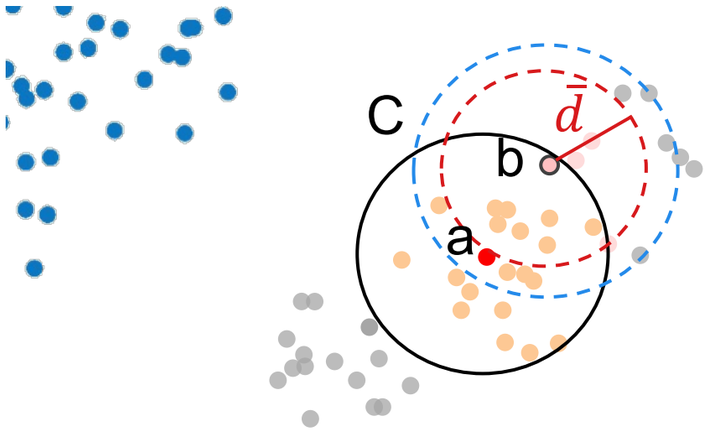
- K-list is the list contains all the k values for LOF algorithm that we will test to determine the inflection point
- k_{inf} is the inflection point in the LOF score curve
- Rare category size is determined by k_{inf}



Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Expanding the annotated category to include points with similar local density as central point

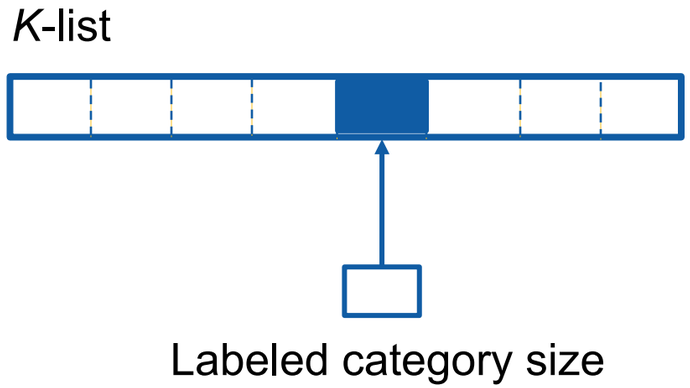


$\bar{d}$ is the averaged distance between the class center and all its neighbors in C

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

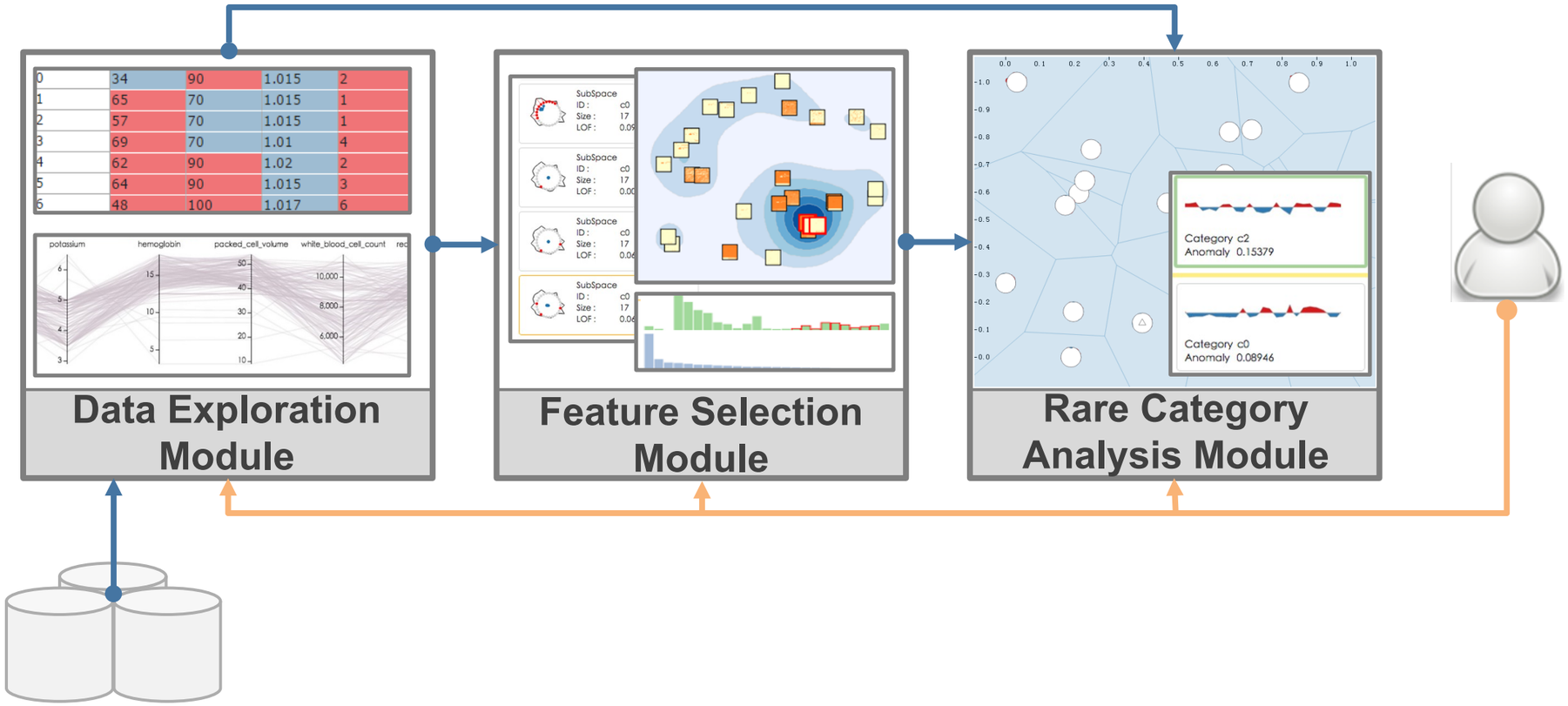
- Refine the granularity of *K*-list based on labeled category size



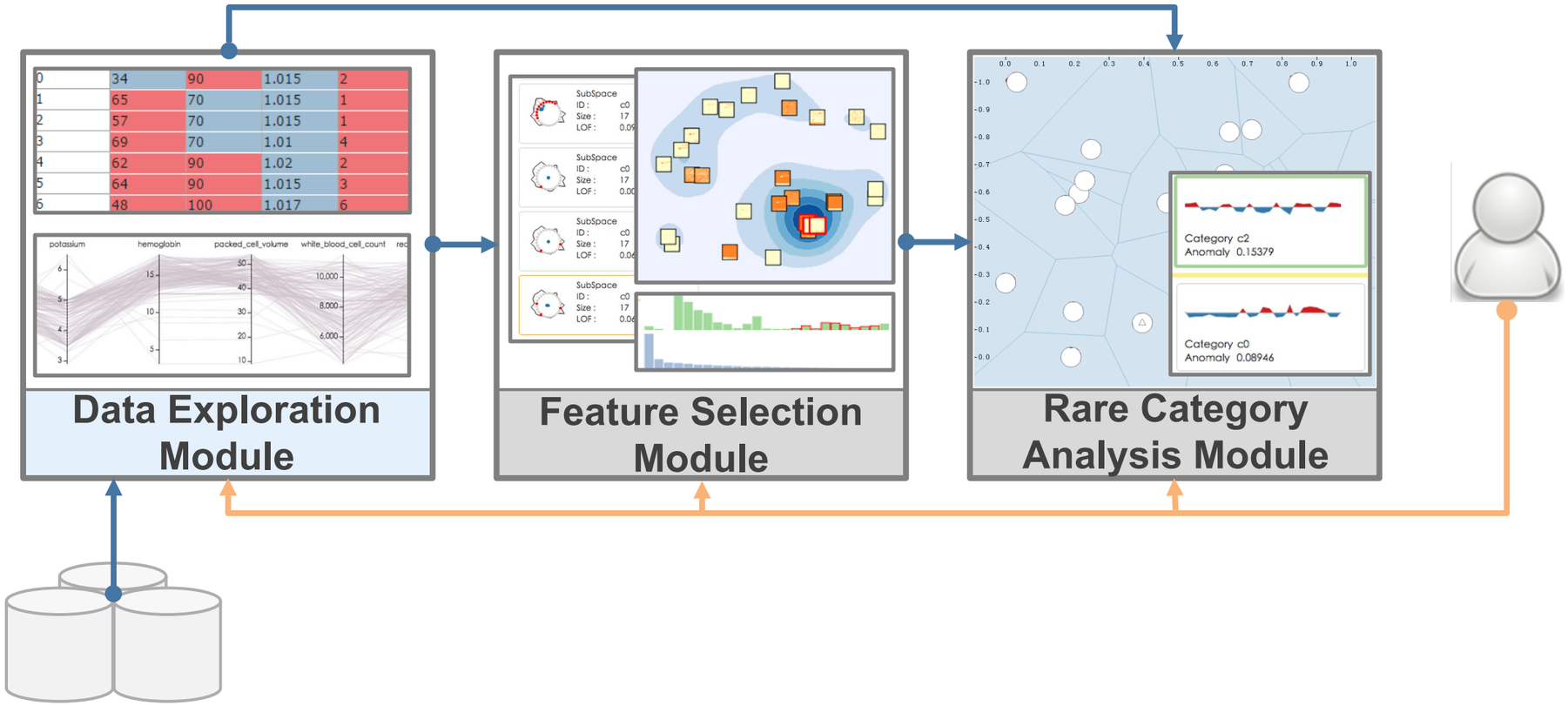
Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

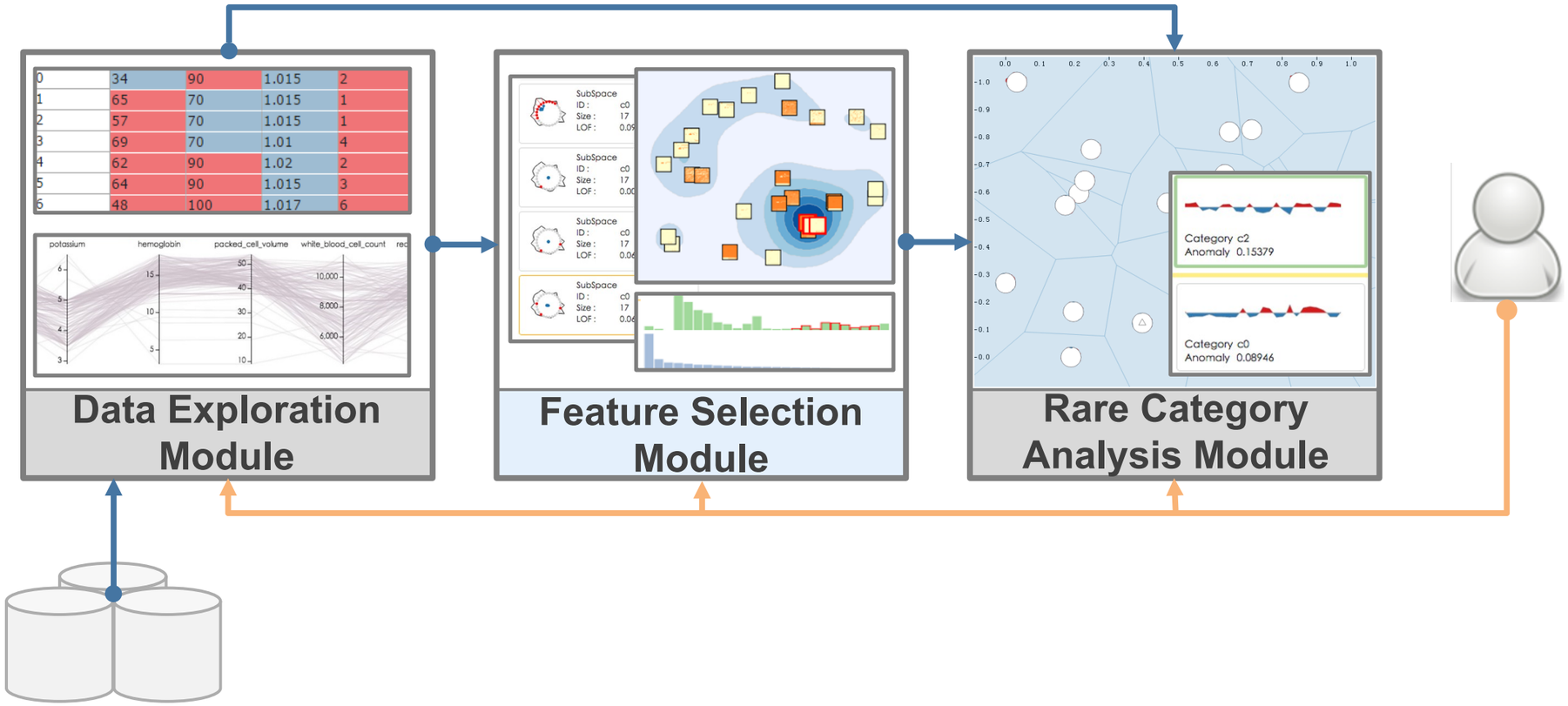
Visualization Design



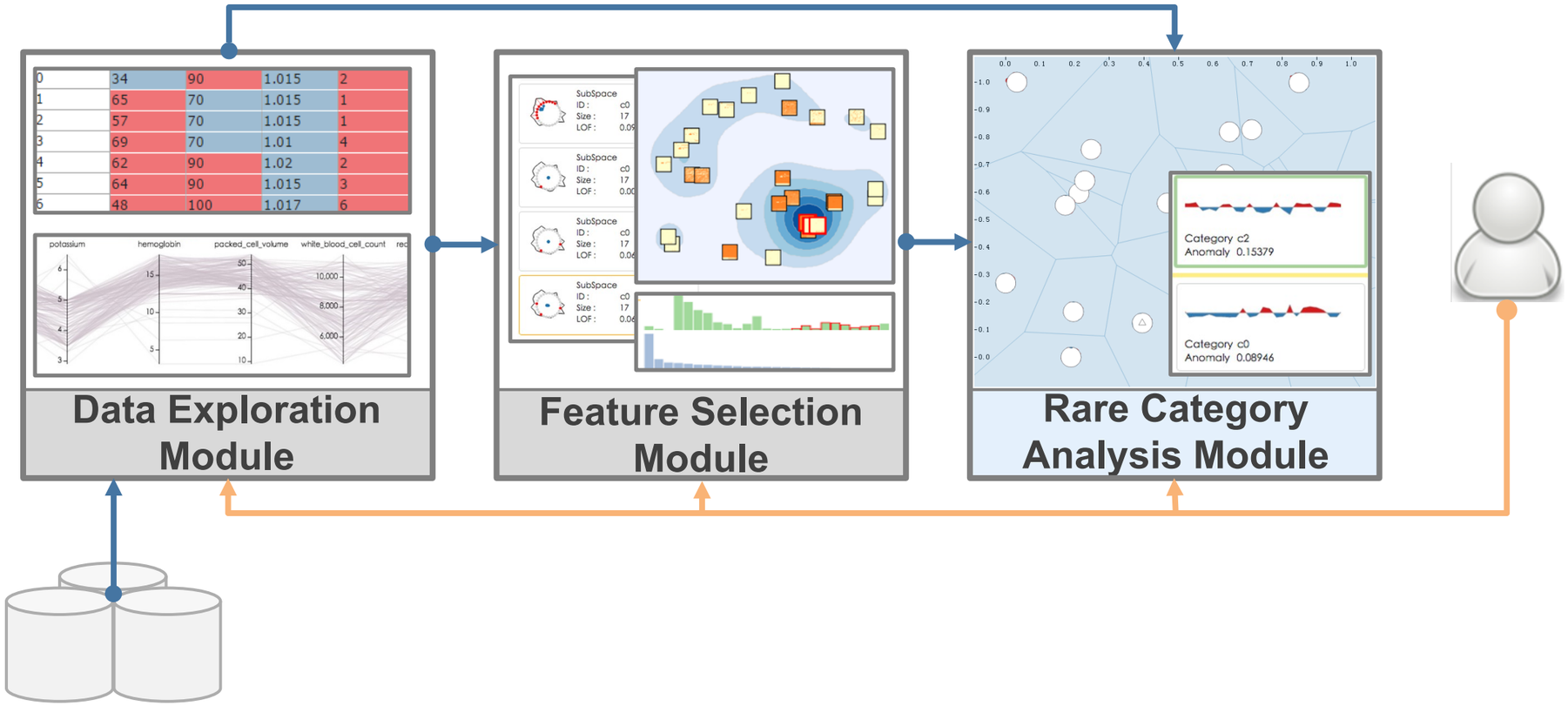
Visualization Design

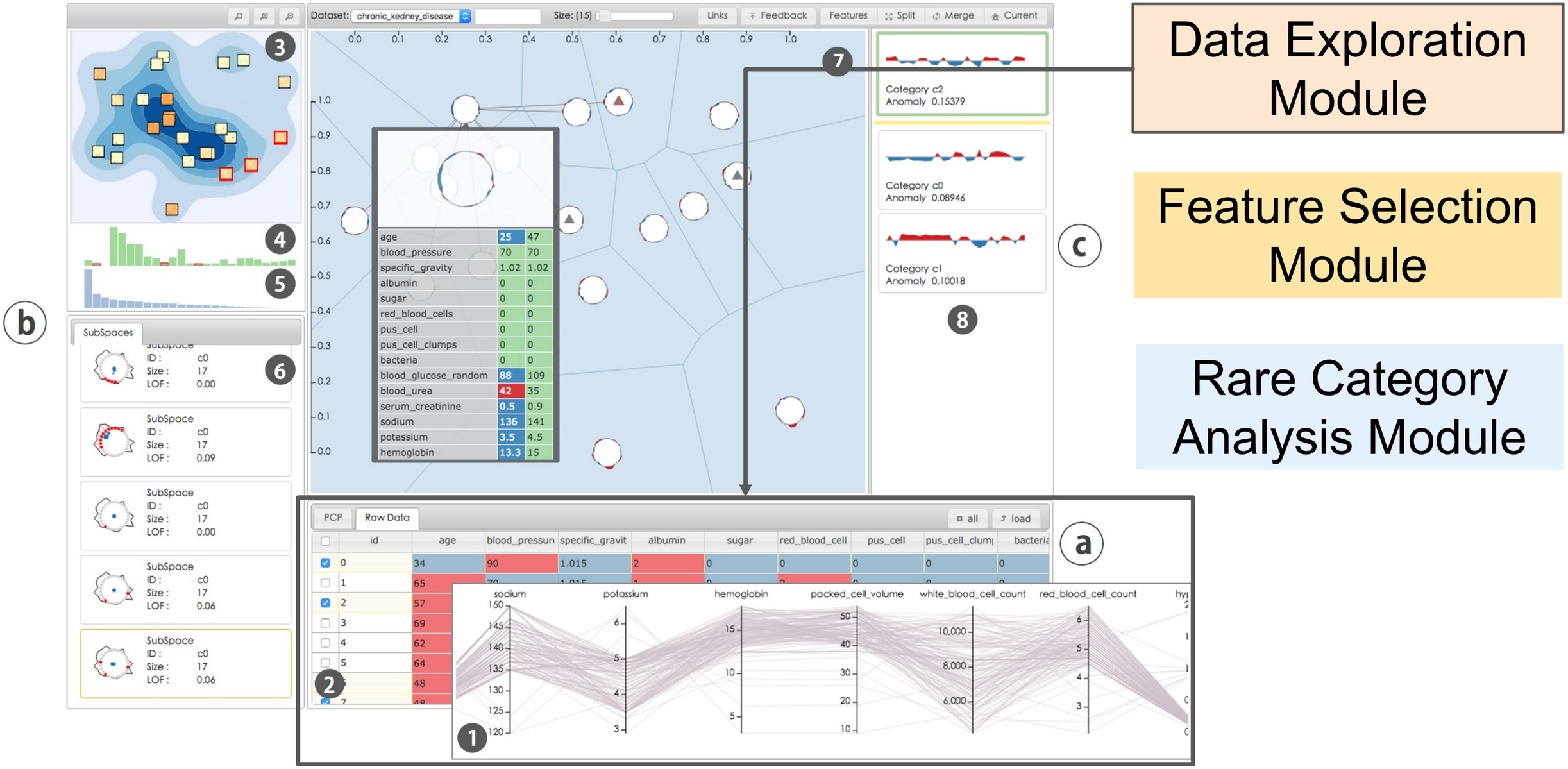


Visualization Design



Visualization Design

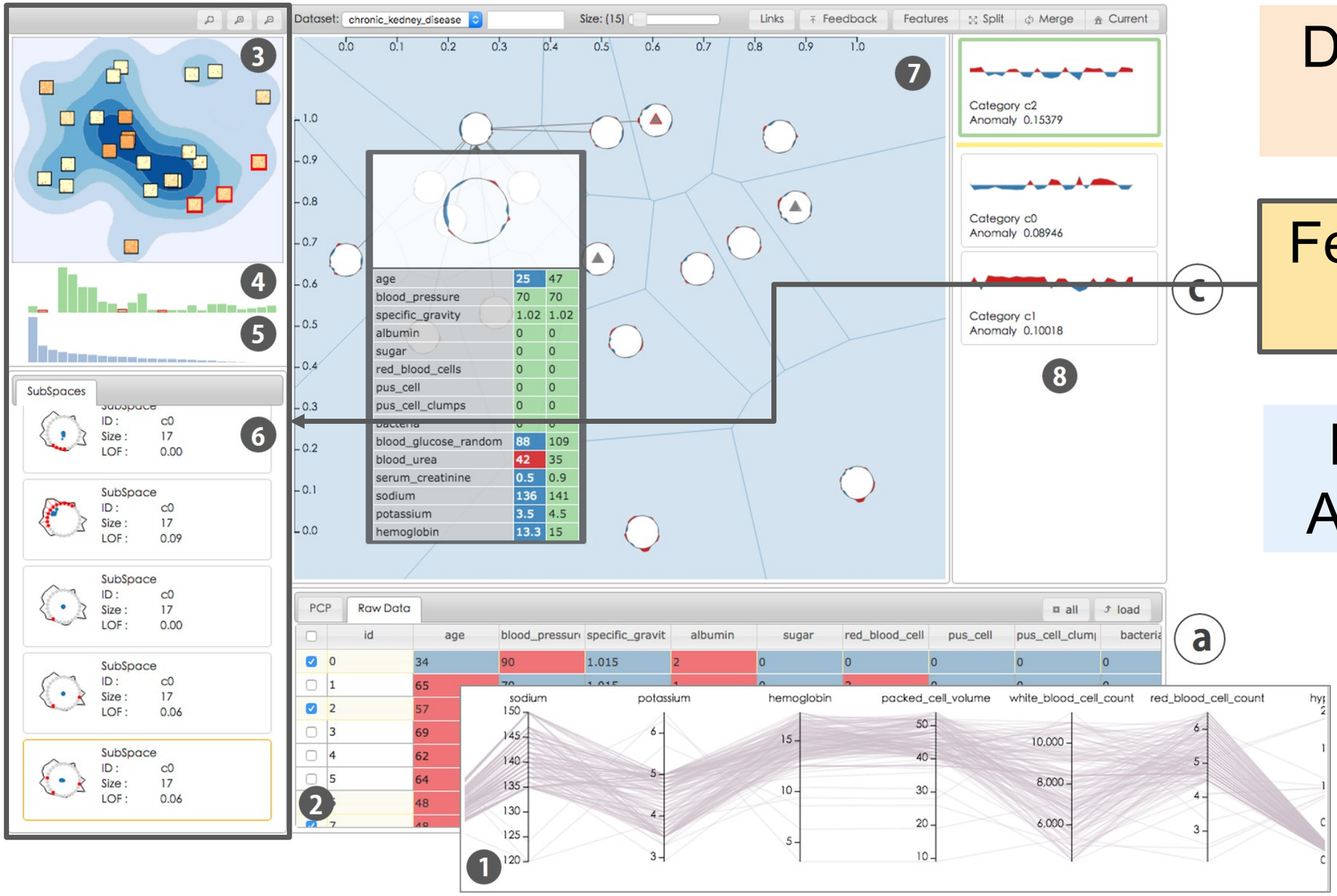




Data Exploration Module

Feature Selection Module

Rare Category Analysis Module

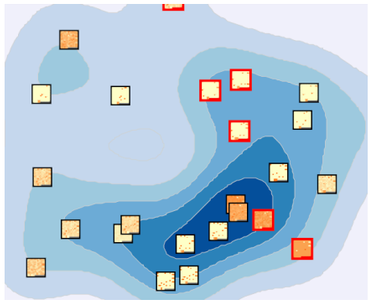


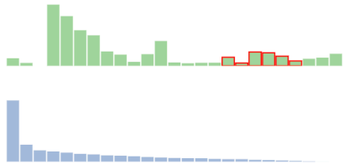
Data Exploration Module

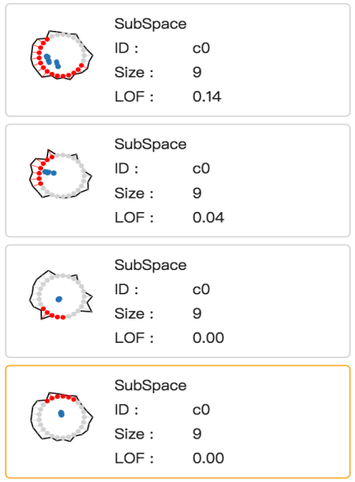
Feature Selection Module

Rare Category Analysis Module

Feature Selection Module



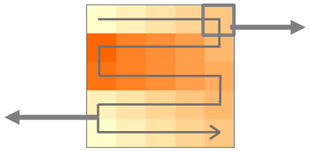




Feature Distribution View

- multidimensional scaling → correlated features are clustered together while independent ones are positioned apart

- feature order



feature value of instances

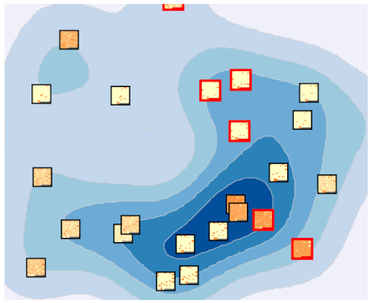
high ⟷ low

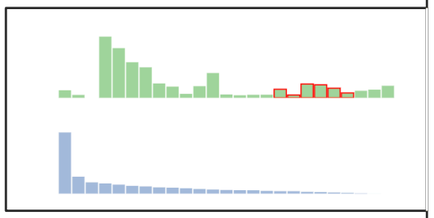
$$min \sum_{i,j} \omega_{ij} ||x_i - x_j||^2$$

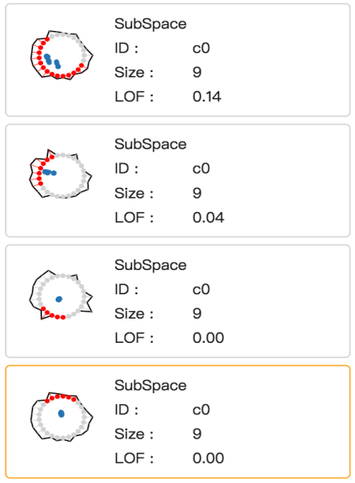
-  uniform color: the variance of the corresponding feature is **lower**

 mottled color: the variance of the corresponding feature is **higher**

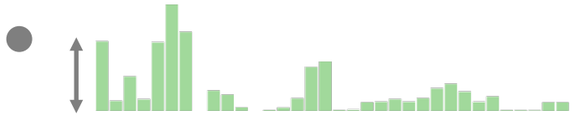
Feature Selection Module







Variance Bar Chart Views



height: feature's variance

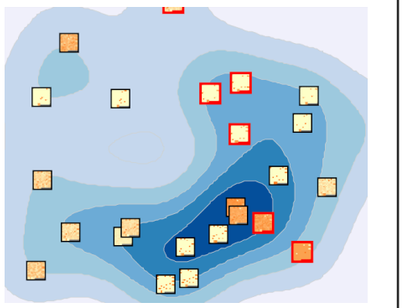
select features

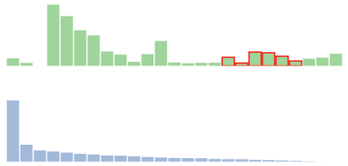
What if individual features are highly correlated and difficult to separate?

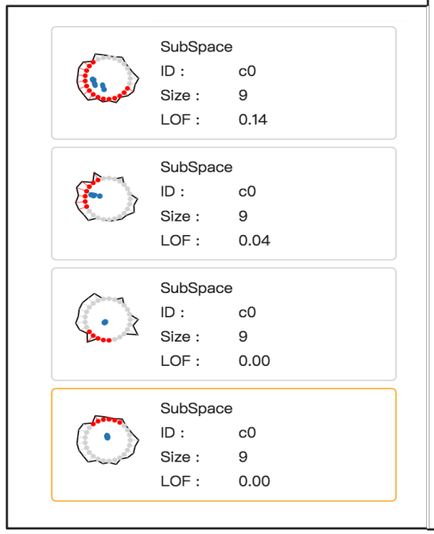
reduce the dimension of features

height: the percentage of the data variances preserved by each principal component calculated by PCA

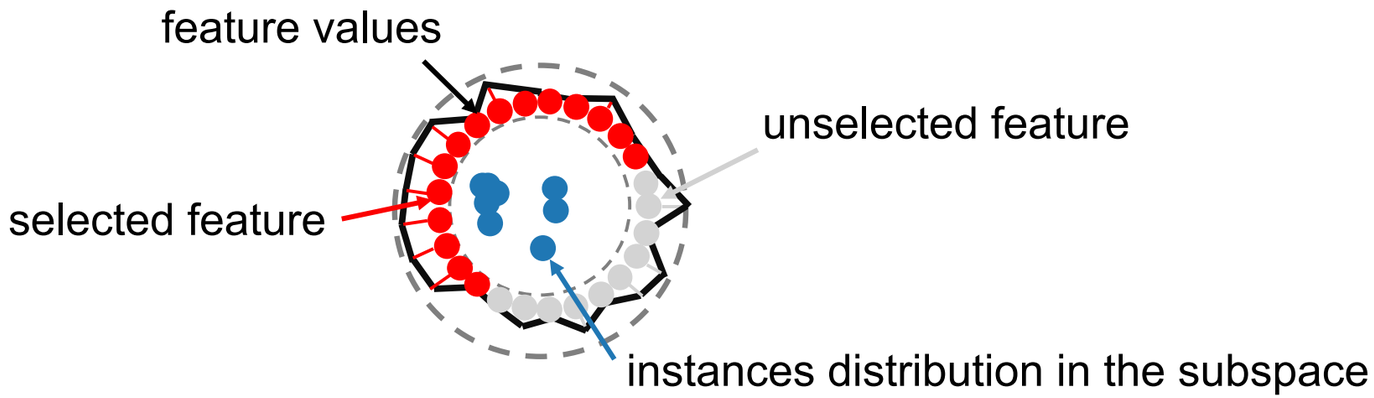
Feature Selection Module

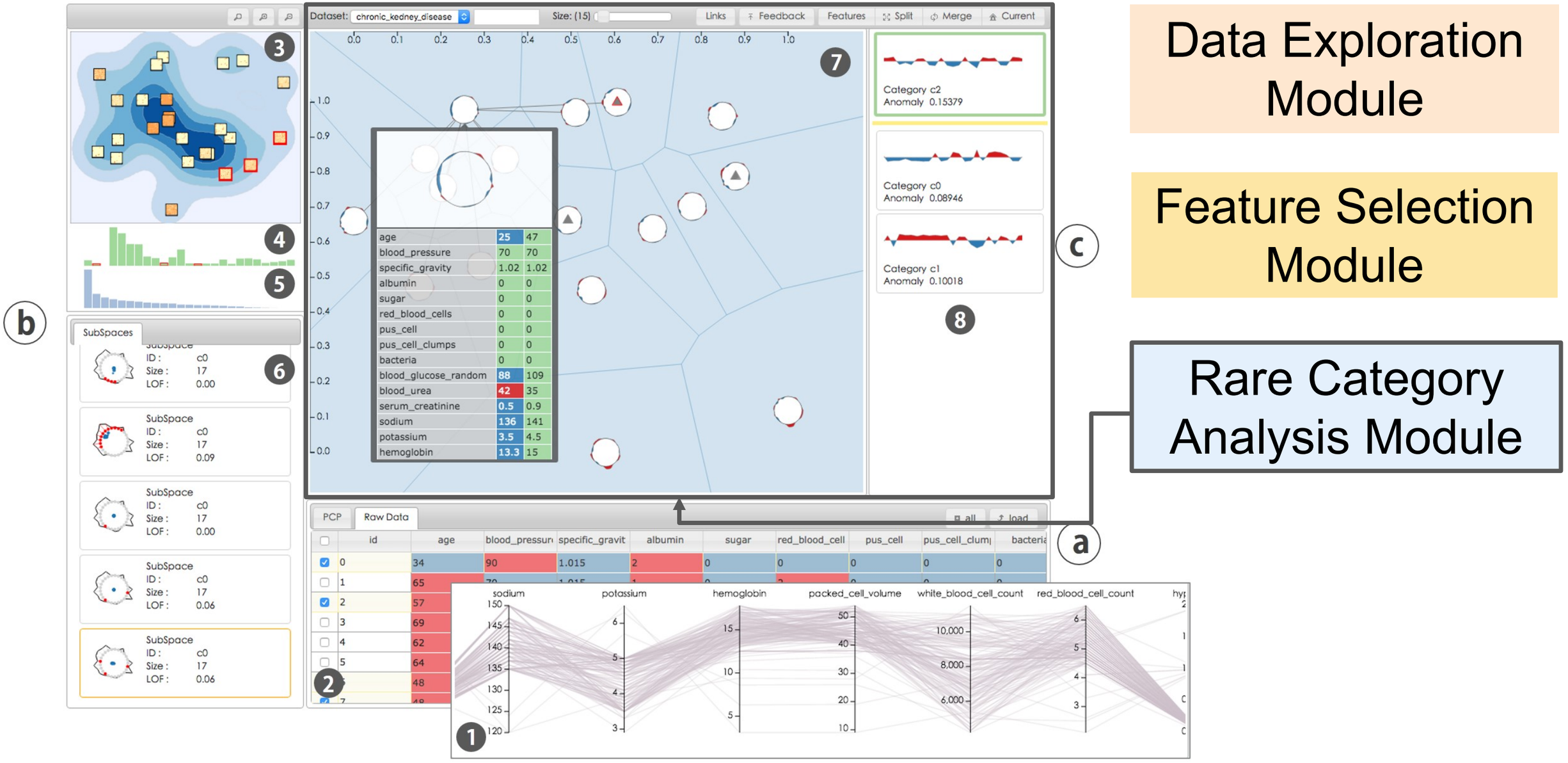






Subspace List



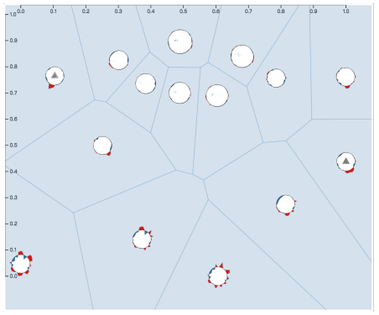


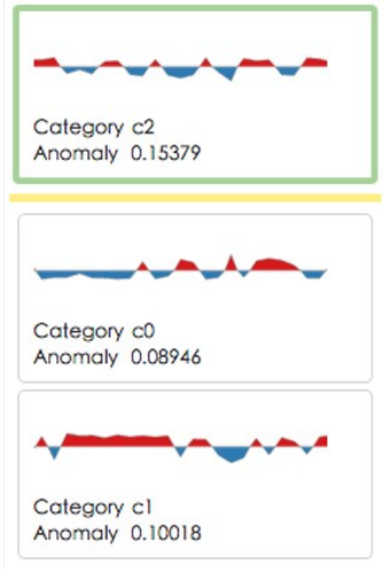
Data Exploration Module

Feature Selection Module

Rare Category Analysis Module

Feature Selection Module

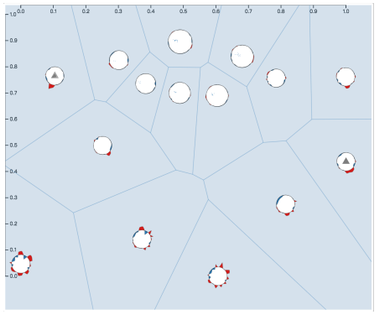


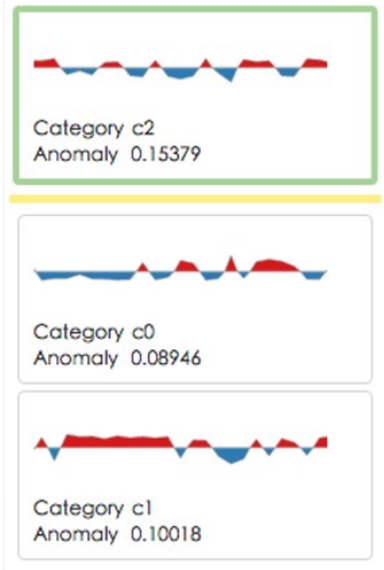


Category View

- multidimensional scaling → k-nearest neighbor relationships

Feature Selection Module



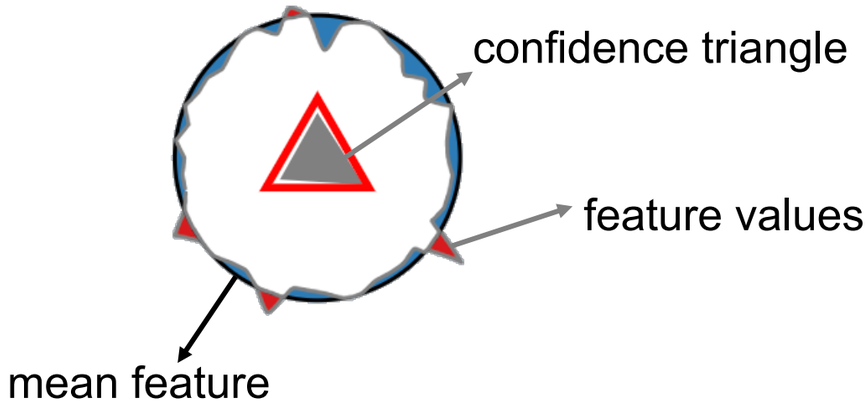


Category View

- Z-glyphs

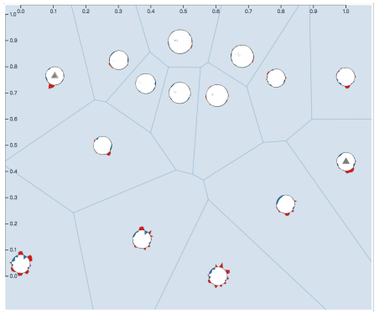
above average

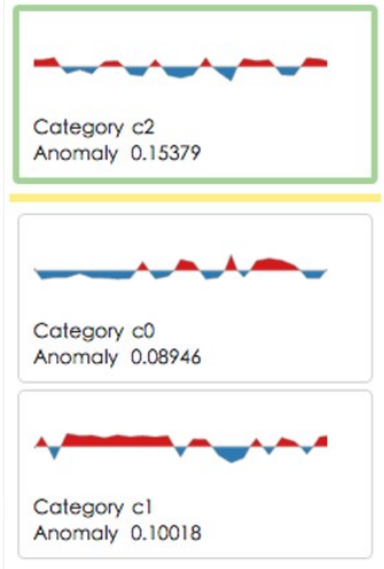
below average



This design helps with the identification of outliers (i.e., the instances having a larger variance compared to the mean) in the focused category.

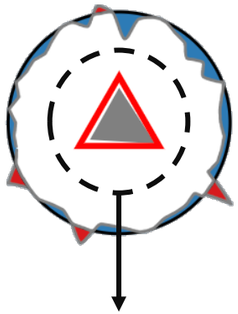
Feature Selection Module



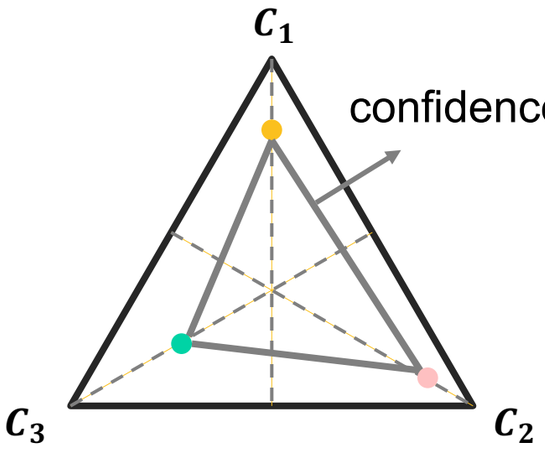


Category View

- Confidence triangle

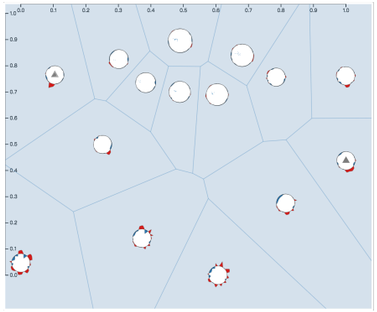


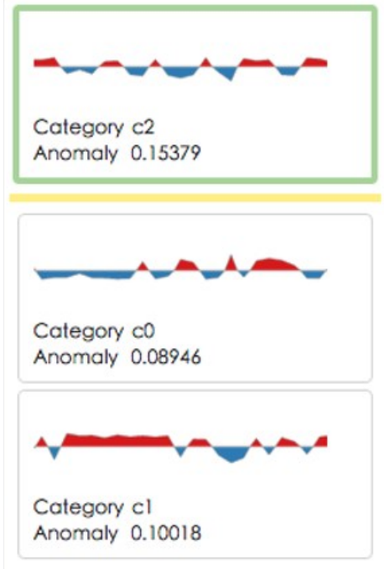
Estimate the confidence of each instance in terms of representing an unknown category from three different criteria.



$$C_{total} = \sqrt[3]{C_1 * C_2 * C_3}$$

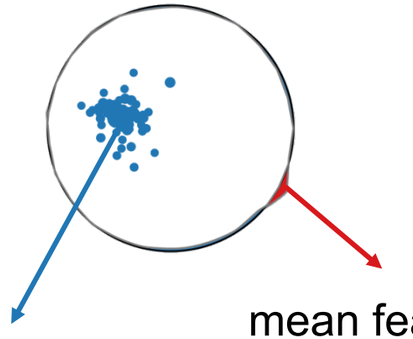
Feature Selection Module





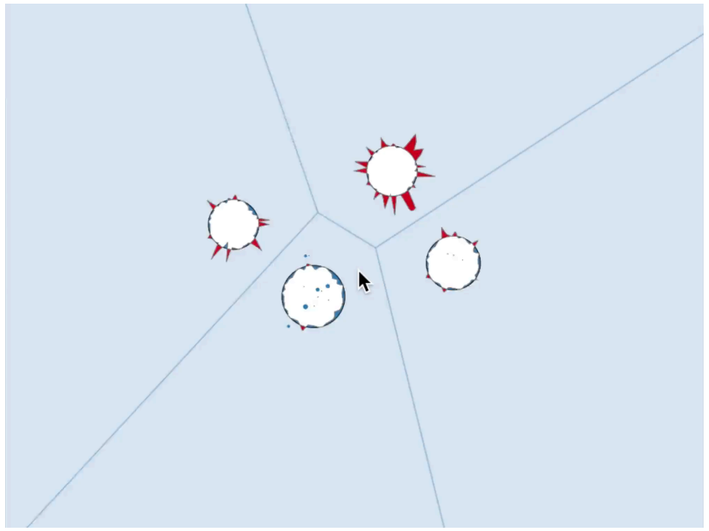
Category View

- a collection glyph for merged circles

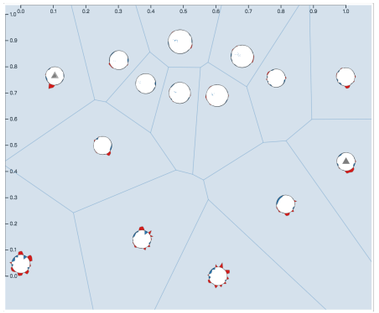


mean feature values

distribution thumbnail



Feature Selection Module

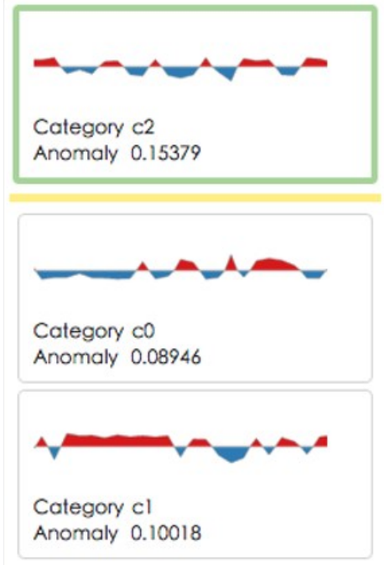


Category List

- A labeled rare category is shown as a horizontal z-glyph arranged in a vertical list.

the mean for feature values over the entire dataset

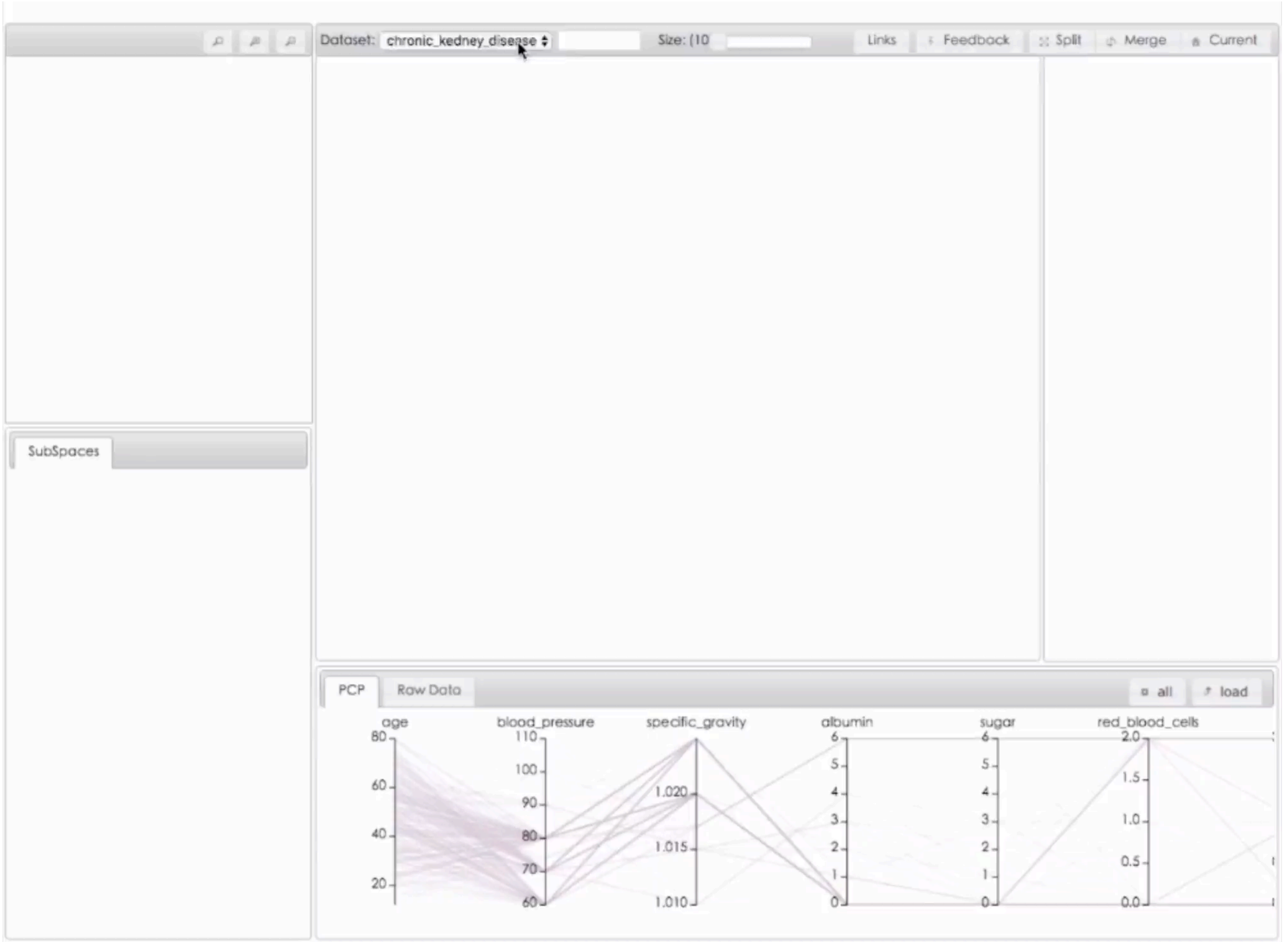
above average
below average



reordered based on their similarities to the focus

users can merge similar categories together to form a single larger category

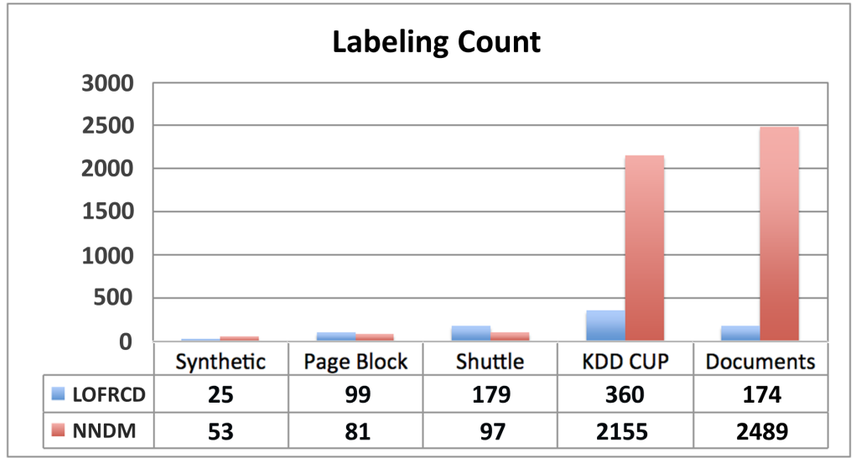
Video Demo


Dataset: chronic_kedney_disease
Size: (10
Links
Feedback
Split
Merge
Current
SubSpaces
PCP
Raw Data
all
load
age
blood_pressure
specific_gravity
albumin
sugar
red_blood_cells
80
60
40
20
110
100
90
80
70
60
1.020
1.015
1.010
6
5
4
3
2
1
0
2.0
1.5
1.0
0.5
0.0

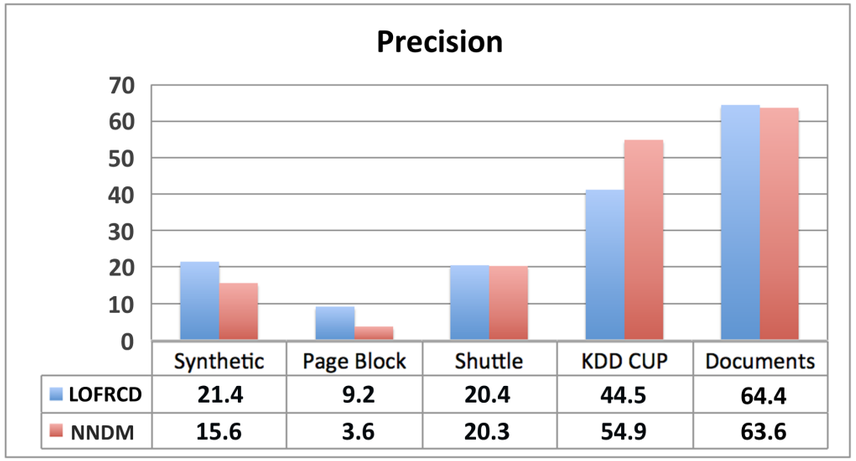
Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

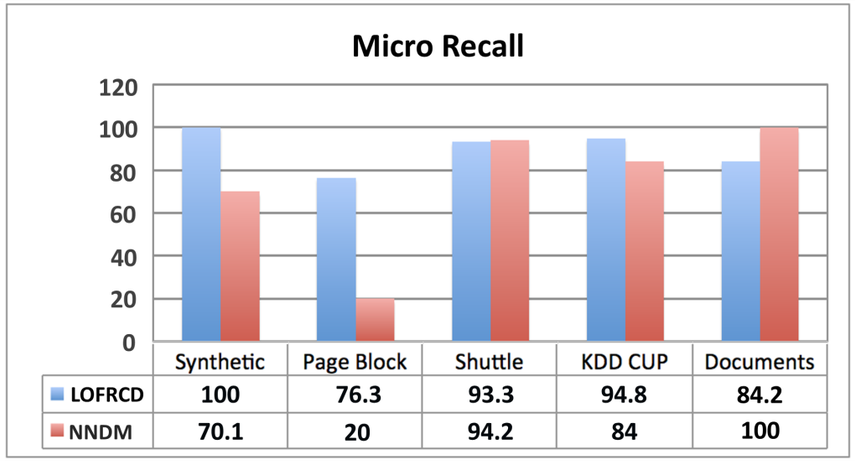
Algorithm Performance



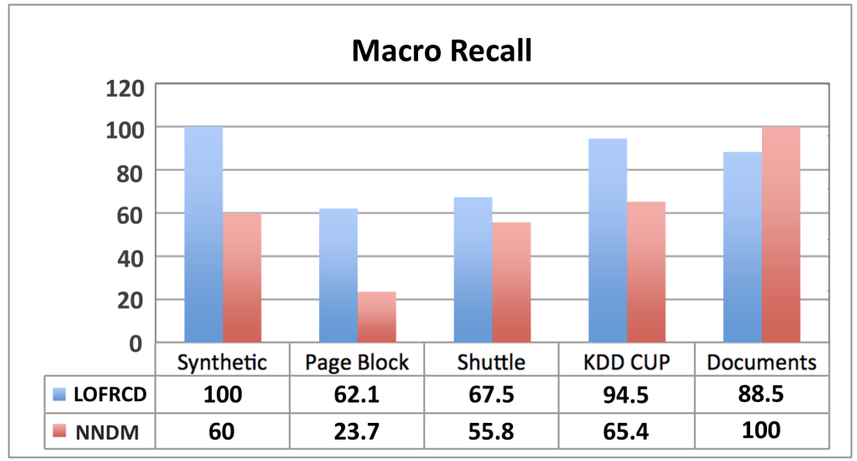
	Synthetic	Page Block	Shuttle	KDD CUP	Documents
LOFRCD	25	99	179	360	174
NNDM	53	81	97	2155	2489



	Synthetic	Page Block	Shuttle	KDD CUP	Documents
LOFRCD	21.4	9.2	20.4	44.5	64.4
NNDM	15.6	3.6	20.3	54.9	63.6



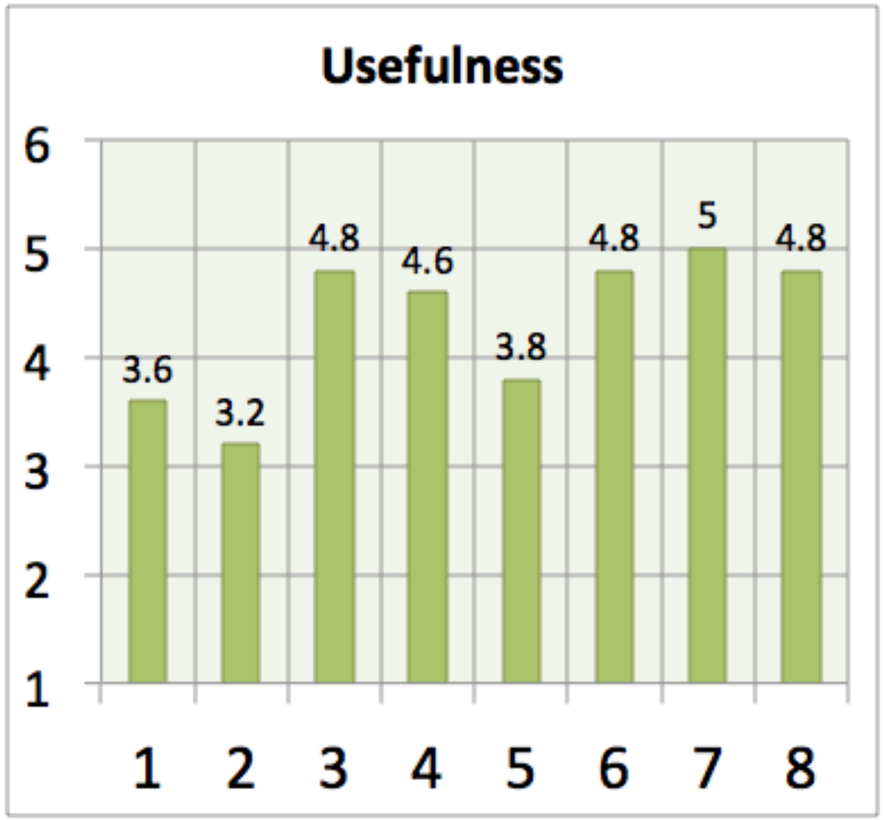
	Synthetic	Page Block	Shuttle	KDD CUP	Documents
LOFRCD	100	76.3	93.3	94.8	84.2
NNDM	70.1	20	94.2	84	100

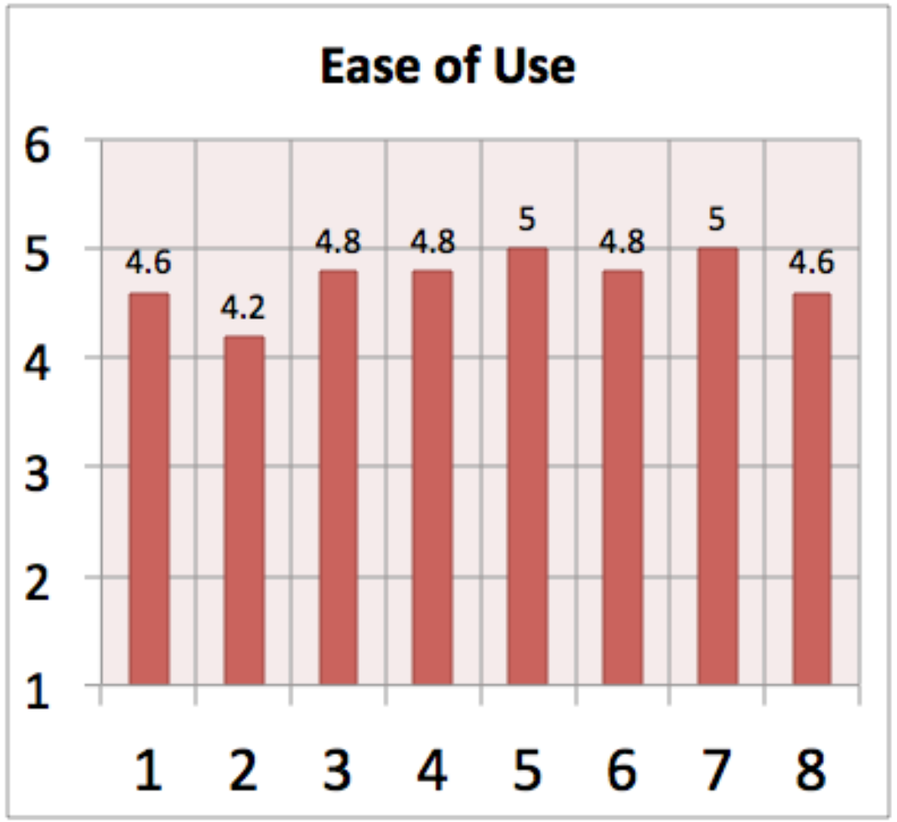


	Synthetic	Page Block	Shuttle	KDD CUP	Documents
LOFRCD	100	62.1	67.5	94.5	88.5
NNDM	60	23.7	55.8	65.4	100

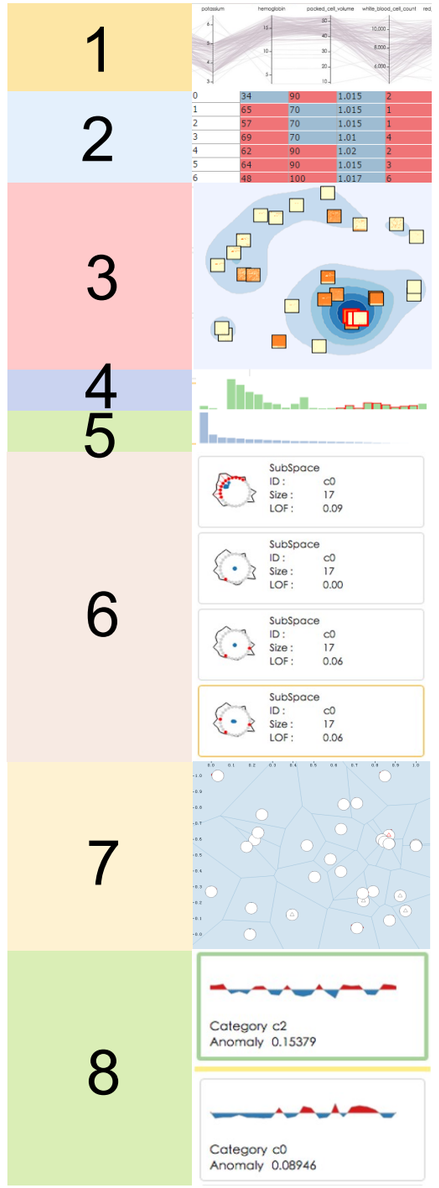
The overall performances of LOFRCD in comparison to NNDM

Case Study





The scores are ranged from 1 (very useless or very difficult to use) 5 (very useful or very easy to use).



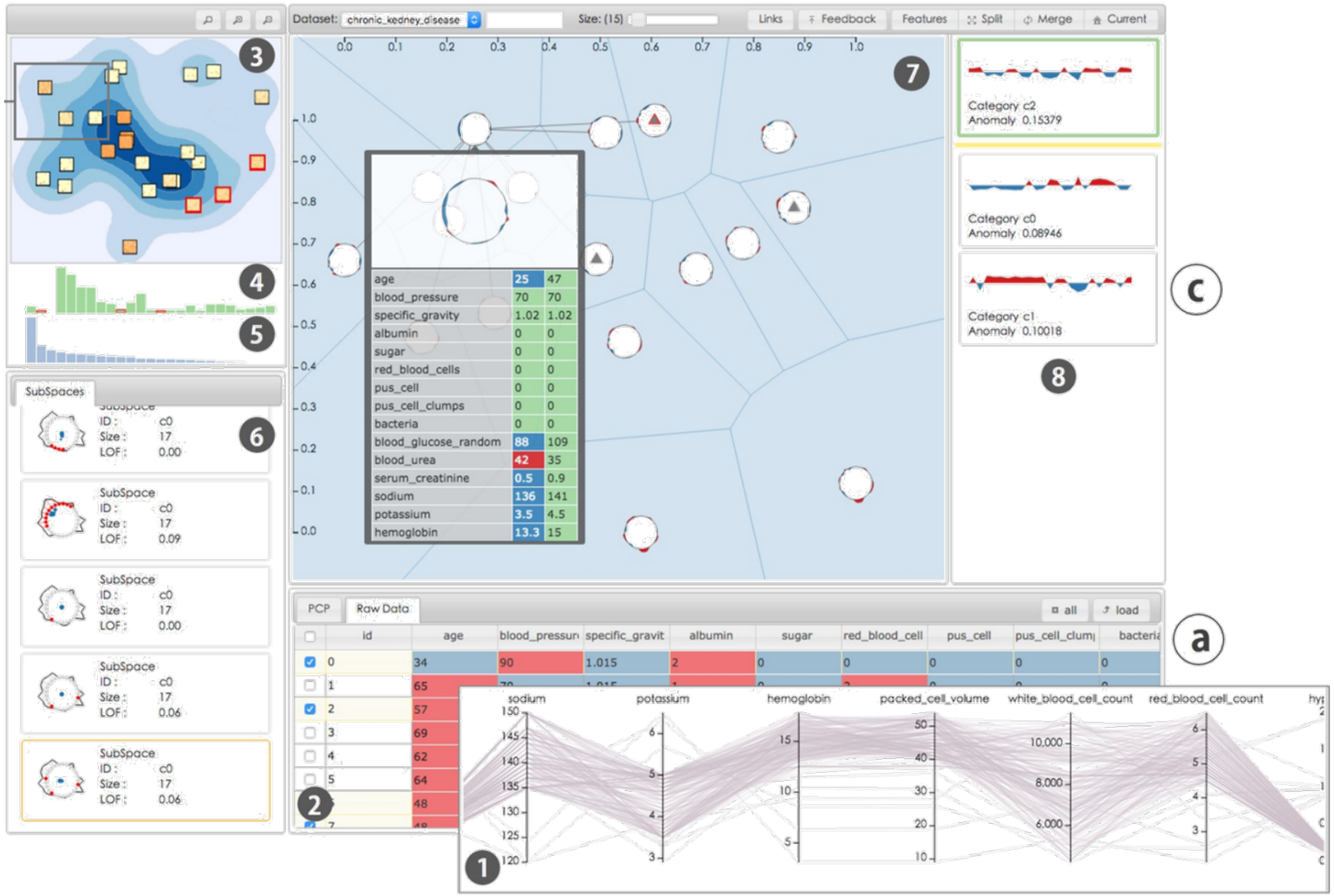
Outline

- Introduction
- LOFRCD Algorithm
- Visualization Design
- Evaluation
- Conclusion

Conclusion

- We introduced RCLens, a visual analytics system designed to support user-guided rare category exploration and identification

- We proposed a novel active learning-based algorithm to iteratively identify more accurate rare categories in response to user-provided feedback.



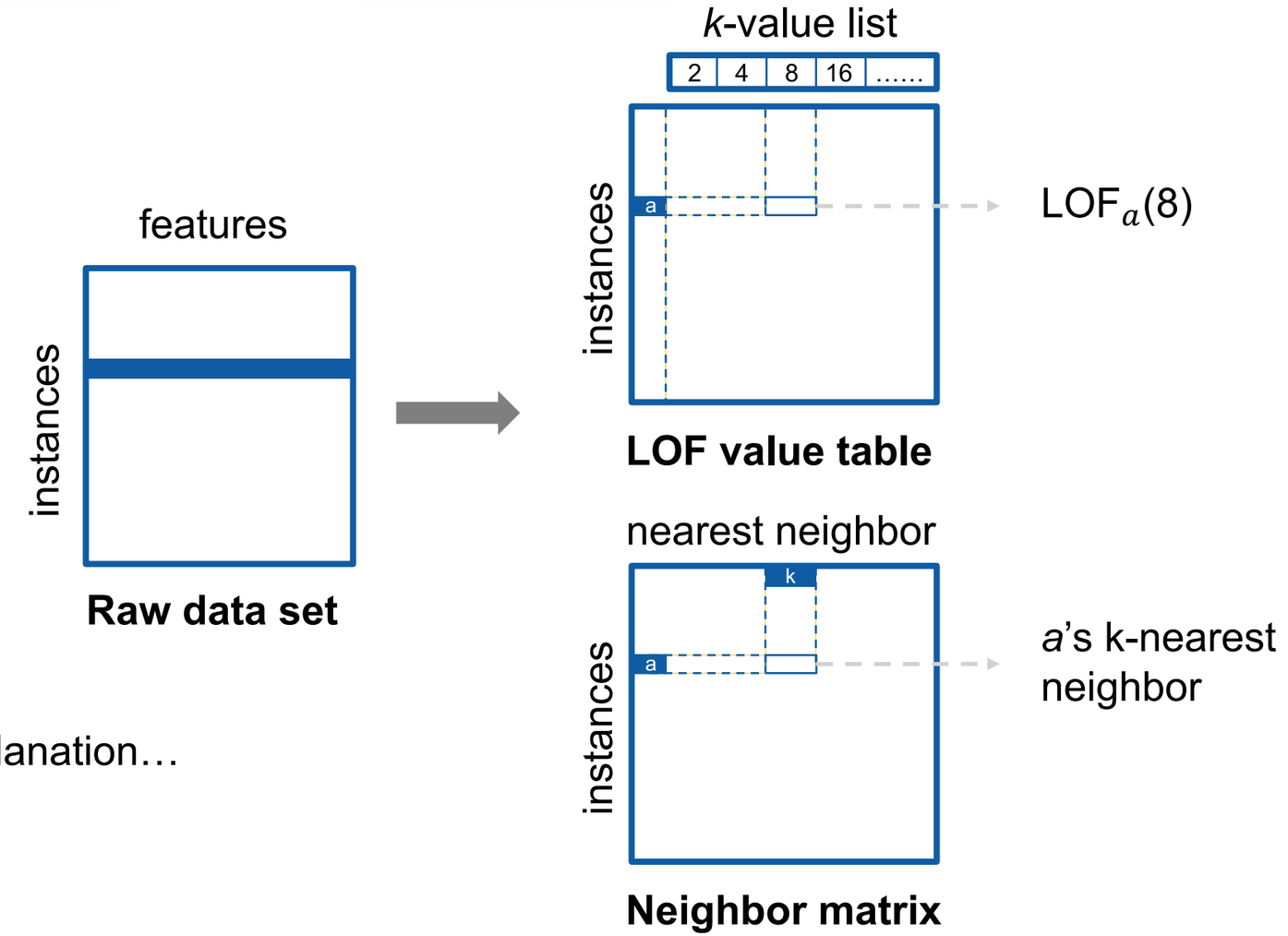
Thank you!

RCLens: Interactive Rare Category Exploration and Identification

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Calculating the LOF score for every *k* in *k*-value list

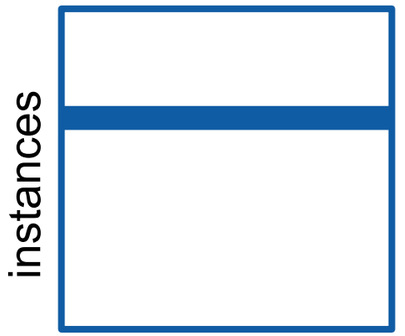


Explanation…

Steps of our algorithm

Step 1: Find representative point → **Step2:** Determine the boundary → **Step3:** Category expansion → **Step4:** Refine hyperparameter

- Calculating the LOF score for every *k* in *k*-value list
- Calculating three confidence scores



LOF value table

C_1 : the isolation of a minority class

C_2 : Be close to all its neighbors within the category, and distant from neighbors outside the category

C_3 : a should have a similar kinf value within the category

Explanation…