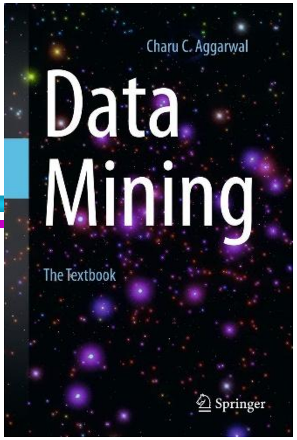
Anomaly Detection



Lecture Notes for Chapter 9

Introduction to Data Mining, 2nd Edition

by

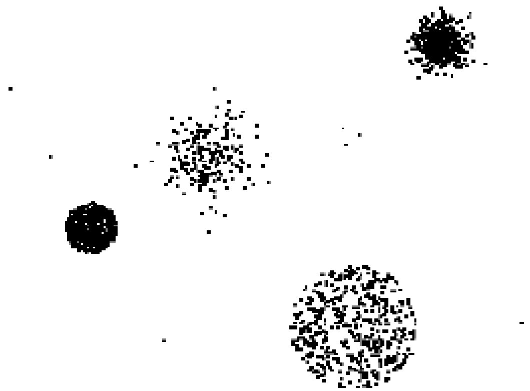
Tan, Steinbach, Karpatne, Kumar

Anomaly/Outlier Detection

- In anomaly detection, the goal is to find objects that do not conform to normal patterns or behavior
  - Often, anomalous objects are known as **outliers**, since, on a scatter plot of the data, they lie far away from other data points
  - Also known as **deviation detection**, because anomalous objects have attribute values that deviate significantly from the expected or typical attribute values
  - Also known as **exception mining**, because anomalies are exceptional in some sense

Anomaly/Outlier Detection

- What are anomalies/outliers?
  - The set of data points that are considerably different than the remainder of the data



- Natural implication is that anomalies are relatively rare
  - One in a thousand occurs often if you have lots of data
  - Context is important, e.g., freezing temps in July

- Can be important or a nuisance
  - Unusually high blood pressure (important)
  - 200 pound, 2 year old (nuisance)

Applications

- Fraud Detection
- Intrusion Detection
- Dropout
- etc.

Anomaly/Outlier Detection

- *Although much of the recent interest in anomaly detection is driven by applications in which anomalies are the focus, historically, anomaly detection (and removal) has been viewed as a data preprocessing technique to eliminate erroneous data objects that may be recorded because of human error, a problem with the measuring device, or the presence of noise*
  - *Such anomalies provide no interesting information but only distort the analysis of normal objects*
- *The emphasis here is on detecting anomalous objects that are interesting in their own right*

A Definition of an Anomaly

- An important characteristic of an anomaly detection problem is the way an anomaly is defined
  - They can be defined in different ways depending on the problem requirements
- The following high-level definition of an anomaly encompasses most of the definitions commonly employed
  - An **anomaly** is an observation that doesn't fit the distribution of the data for normal instances, i.e., is unlikely under the distribution of the majority of instances

A Definition of an Anomaly

- The definition does not assume that the distribution is easy to express in terms of well-known statistical distributions
  - The difficulty of doing so is the reason that many anomaly detection approaches use non-statistical approaches

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Univariate or Multivariate
  - Record Data or Proximity Matrix
  - Availability of Labels
  - Relatively Small in Number
  - Sparsely Distributed

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Univariate or Multivariate
    - ◆ Single atribute: an object is anomalous depends on whether the object's value for that attribute is anomalous
    - ◆ Multiple atributes = a data object may have anomalous values for some attributes but ordinary values for other attributes
      - An object may be anomalous even if none of its attribute values are individually anomalous
        - It is common to have people who are two feet tall (children) or are 100 pounds in weight, but uncommon to have a two-feet tall person who weighs 100 pounds

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Record Data or Proximity Matrix
    - For the purpose of anomaly detection, it is often sufficient to know how different an instance is in comparison to other instances
    - Hence, some anomaly detection methods work with a different representation of the input data
      - Proximity matrix = every entry in the matrix denotes the pairwise proximity (similarity or dissimilarity) between two instances

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Availability of Labels
    - If the problem translates to a **supervised** learning (classification) problem or not, i.e., **unsupervised**
      - All anomaly detection methods presented here operate in the unsupervised setting

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Relatively Small in Number
    - ◆ Since anomalies are infrequent, most input data sets have a predominance of normal instances
    - ◆ The input data set is thus often used as an imperfect representation of the normal class in most anomaly detection techniques
    - ◆ Some anomaly detection methods also provide a mechanism to specify the expected number of outliers in the input data
      - Such methods can work with a larger number of anomalies in the data

Nature of Data

- The nature of the input data plays a key role in deciding the choice of a suitable anomaly detection technique
  - Sparsely Distributed
    - Anomalies, unlike normal objects, are often unrelated to each other and hence distributed sparsely in the space of attributes
    - The successful operation of most anomaly detection methods depends on anomalies not being tightly clustered
    - However, some anomaly detection methods are specifically designed to find clustered anomalies, which are assumed to either be small in size or distant from other instances

Characteristics of Anomaly Detection Methods

□ Characteristics of anomaly detection methods that are helpful in understanding their commonalities and diferences

- Model-based vs. Model-free
- Global vs. Local Perspective
- Label vs. Score

Model-based vs. Model-free

- Model-based Approaches
  - Model can be parametric or non-parametric
  - Anomalies are those points that don’t fit well
  - Anomalies are those points that distort the model
- Model-free Approaches
  - Anomalies are identified directly from the data without building a model
  - Do not explicitly characterize the distribution of the normal or anomalous classes
  - They are often intuitive and simple to use

Global vs. Local

- An instance can be identified as an anomaly either by considering the global context, e.g., by building a model over all normal instances and using this global model for anomaly detection, or by considering the local perspective of every data instance

Label vs. Score

- Some anomaly detection techniques provide only a binary categorization

- Other approaches measure the degree to which an object is an anomaly
  - This allows objects to be ranked
  - Scores can also have associated meaning (e.g., statistical significance)

Anomaly Detection Techniques

- Statistical Approaches
- Proximity-based (Distance-based)
  - Anomalies are points far away from other points
- Clustering-based
  - Points far away from cluster centers are outliers
  - Small clusters are outliers
- Reconstruction-based
- One-class Classification
- Information Theoretic Approaches

Proximity/Distance-based Approaches

- Anomalies are those instances that are most distant from the other objects
  - Assumption = normal instances are related and hence appear close to each other, while anomalies are different from the other instances and hence are relatively far from other instances
- Since the techniques are based on distances, they are also referred to as **distance-based outlier detection techniques**

Distance-based Approaches

- They are model-free anomaly detection techniques
- They make use of the local perspective of every data instance to compute its anomaly score
- They are more general than statistical approaches
  - It is often easier to determine a meaningful proximity measure for a data set than to determine its statistical distribution

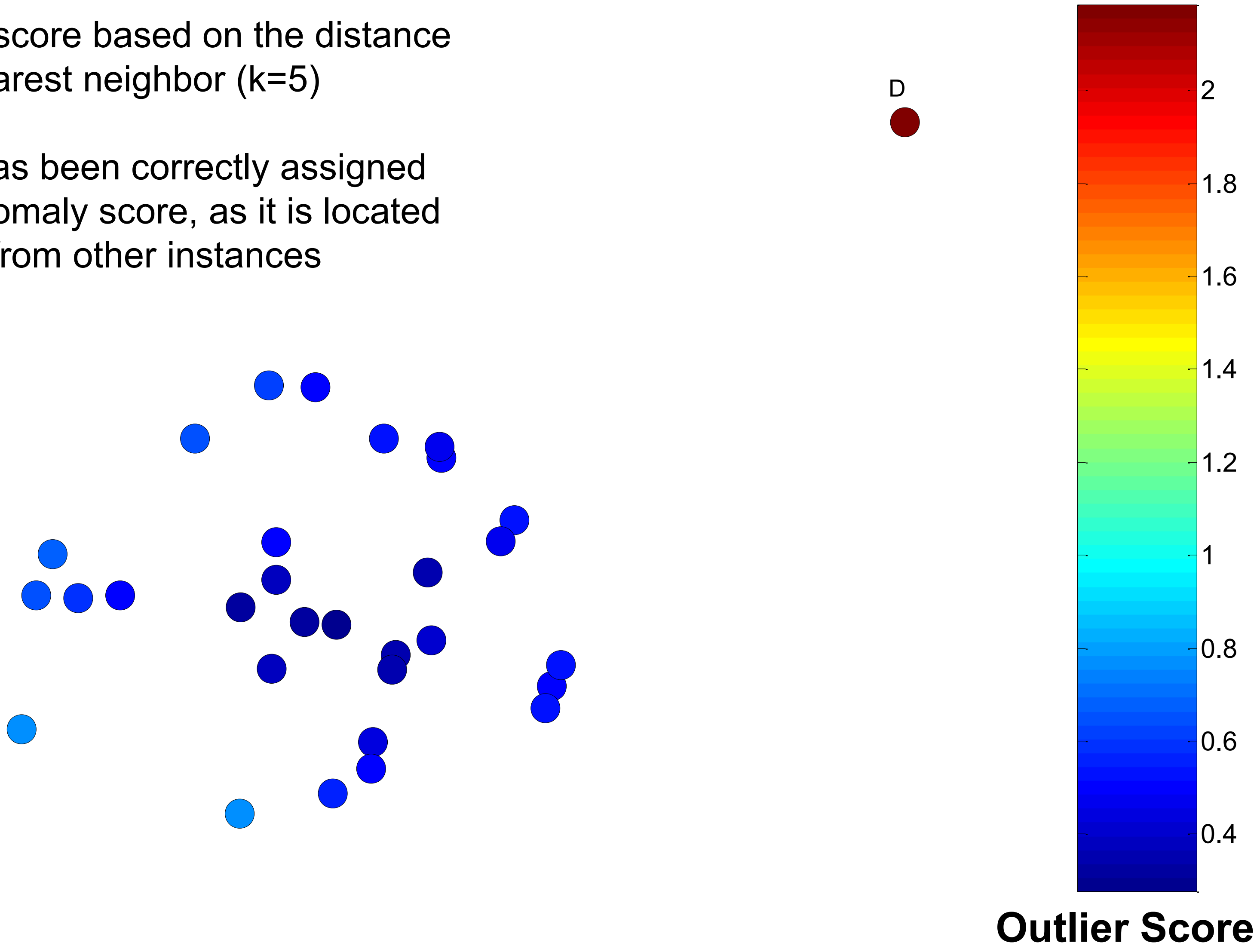
1. Distance-based Anomaly Score

- One of the simplest ways
- The outlier score of an object **x** is the distance to its kth nearest neighbor, dist(**x**, k)
  - Anomalous instance **x** will be quite distant from its k-neighboring instances and would thus have a high value of dist(**x**, k)

Distance-based Anomaly Score

Anomaly score based on the distance to fifth nearest neighbor (k=5)

Point D has been correctly assigned a high anomaly score, as it is located far away from other instances

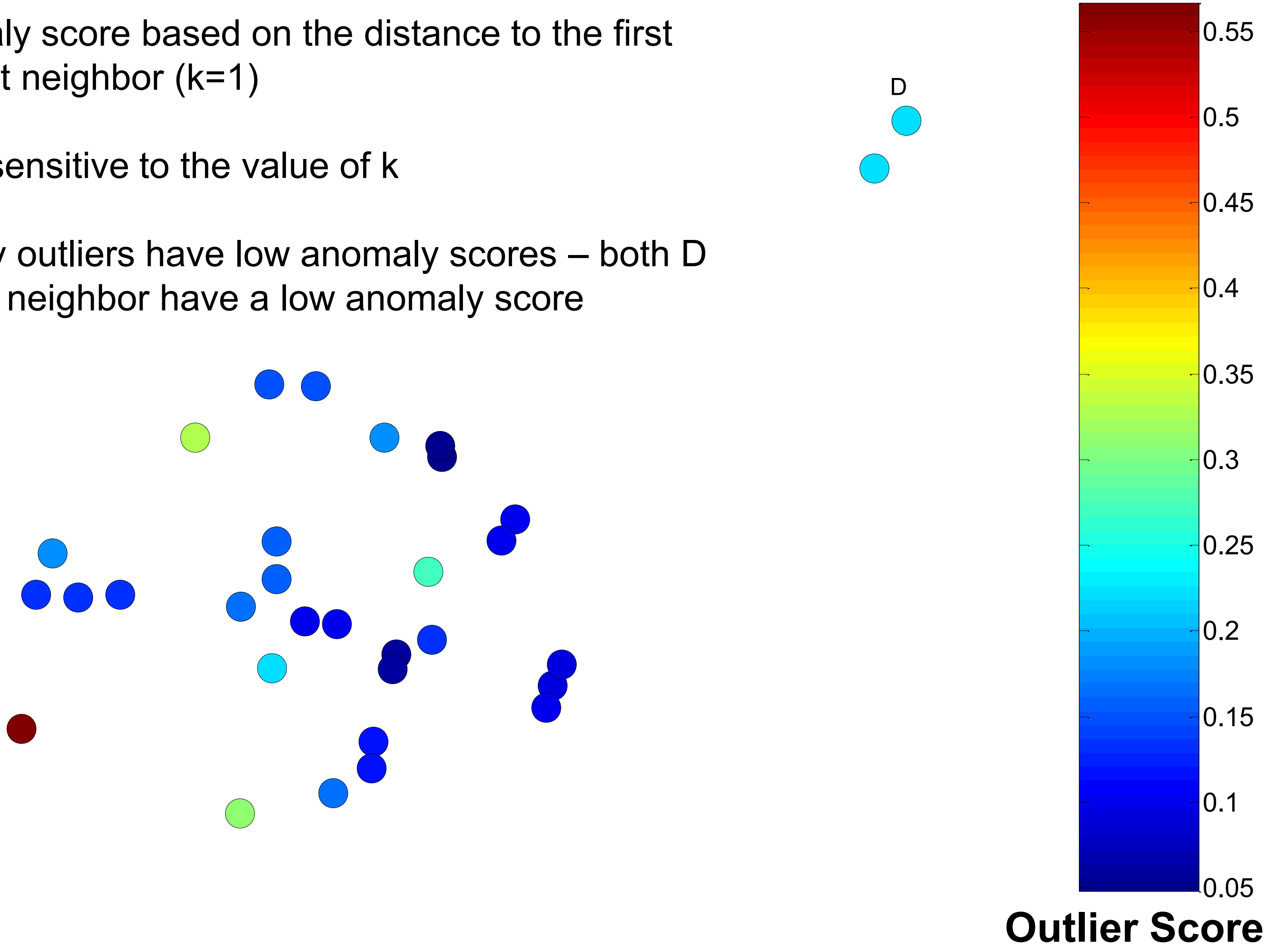


Distance-based Anomaly Score

Anomaly score based on the distance to the first nearest neighbor (k=1)

Quite sensitive to the value of k

Nearby outliers have low anomaly scores – both D and its neighbor have a low anomaly score

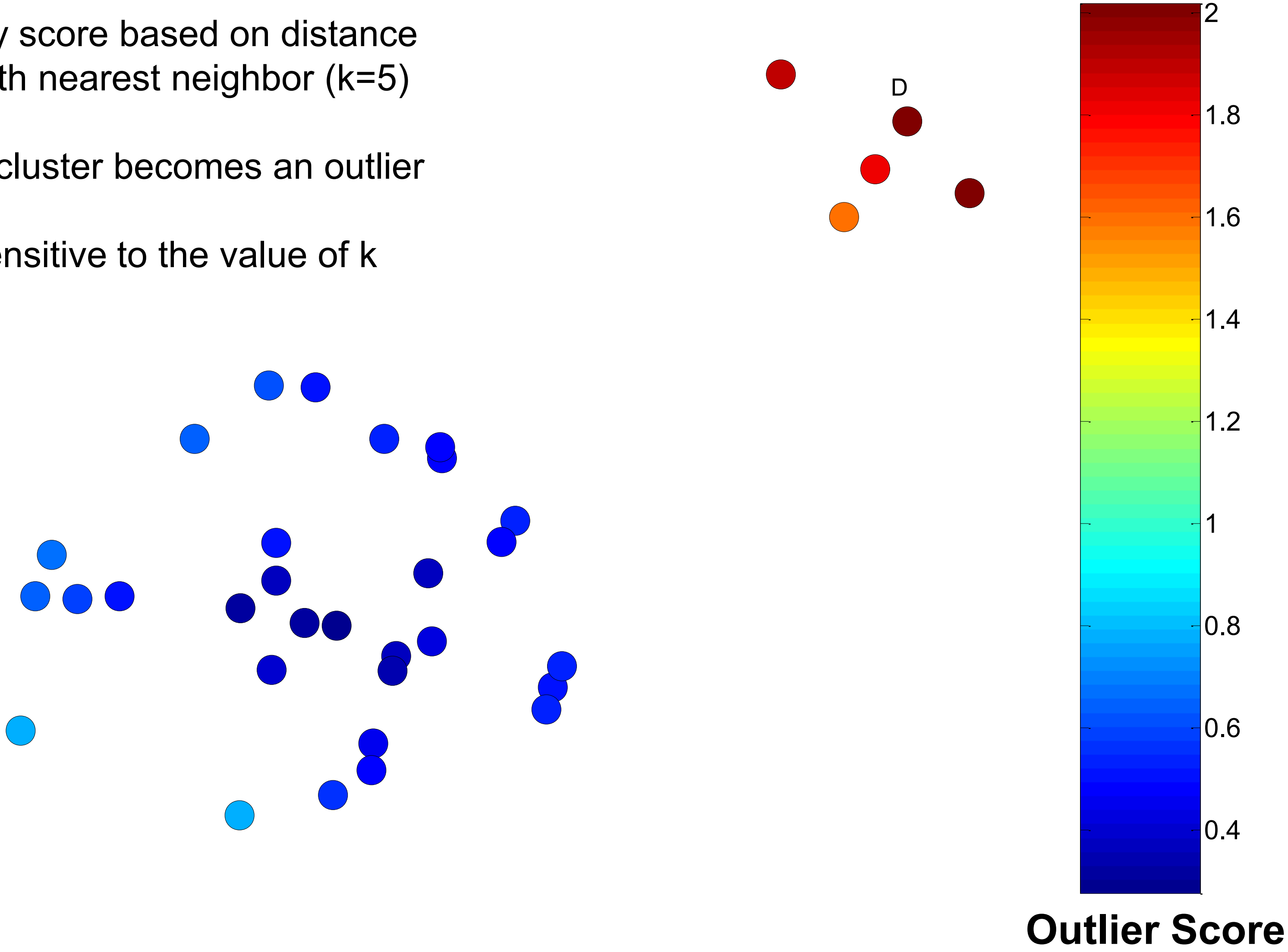


Distance-based Anomaly Score

Anomaly score based on distance to the fifth nearest neighbor (k=5)

A small cluster becomes an outlier

Quite sensitive to the value of k



Distance-based Anomaly Score

- Quite sensitive to the value of k
  - An alternative score that is more robust to the choice of k is the average distance to the first k-nearest neighbors, avg.dist(**x**, k)
    - Widely used in a number of applications as a reliable score

2. Density-based Anomaly Score

- Anomalies are instances that are in regions of low density
- Density = number of instances within a specified distance d from the instance
  - An anomaly will have a smaller number of instances within a distance d than a normal instance
  - Definition similar to the one used by the **DBSCAN**
- Sensitive to parameter d
  - If d is too small, then many normal instances can incorrectly show low density values
  - If d is too large, then many anomalies may have densities that are similar to normal instances

Density-based Anomaly Score

- The distance-based and density-based views of proximity are quite similar to each other
  - dist(**x**, k) provides a measure of the density around **x**, using a different value of d for every instance
    - If dist(**x**, k) is large, the density around **x** is small, and vice-versa
  - Distance-based and density-based anomaly scores thus follow an inverse relationship

$$
\boxed{\begin{aligned}
density(\mathbf{x}, k) &= 1/dist(\mathbf{x}, k),\\
avg.density(\mathbf{x}, k) &= 1/avg.dist(\mathbf{x}, k).
\end{aligned}}
$$

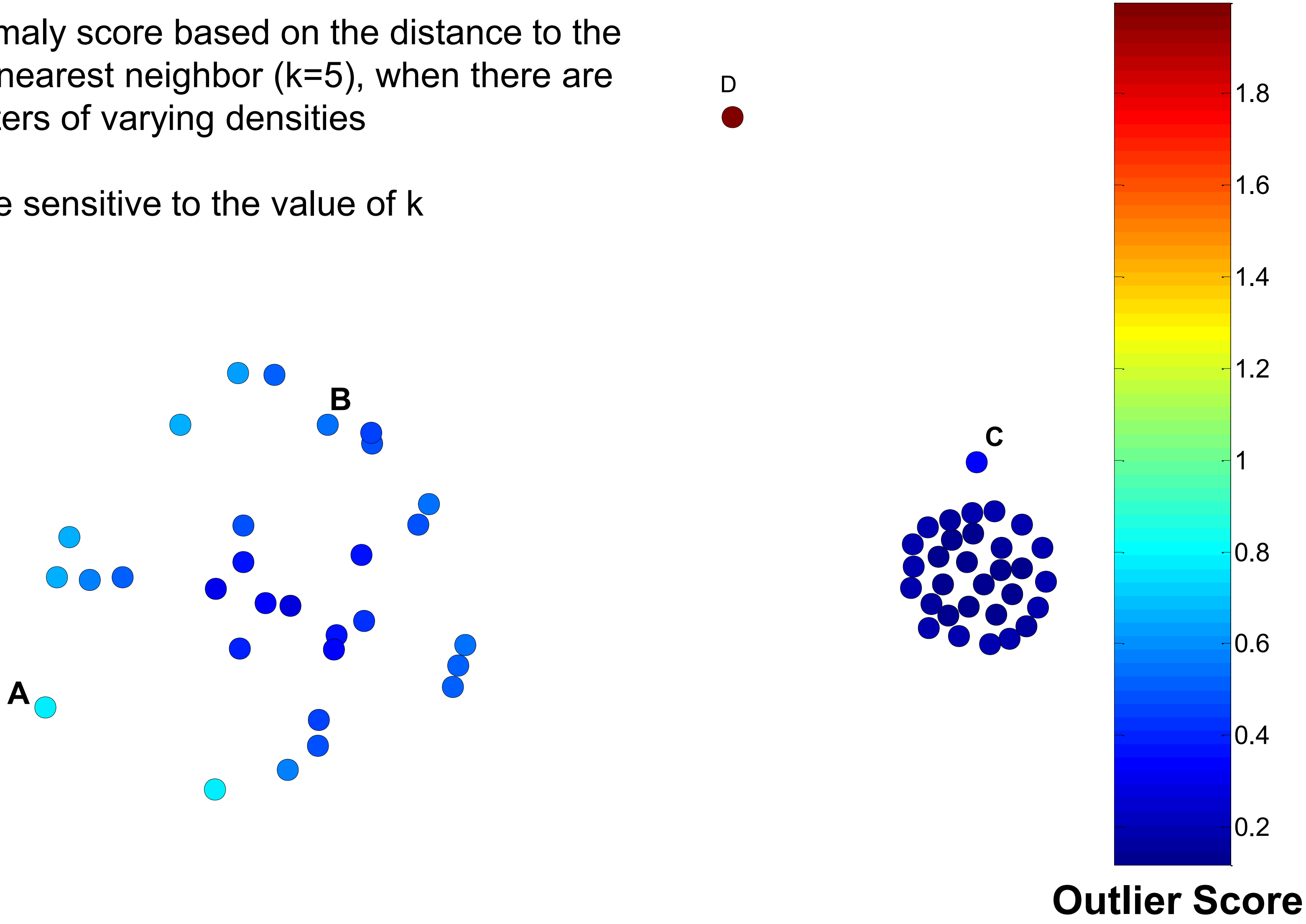
3. Relative Density-based Anomaly Score

- The previous approaches only consider the locality of an individual instance for computing its anomaly score
- In scenarios where the data contains regions of varying densities, such methods would not be able to correctly identify anomalies, as the notion of a normal locality would change across regions

Relative Density-based Anomaly Score

Anomaly score based on the distance to the fifth nearest neighbor (k=5), when there are clusters of varying densities

Quite sensitive to the value of k



Relative Density-based Anomaly Score

- To correctly identify anomalies in such data sets, it is necessary a notion of density that is relative to the densities of neighboring instances
  - One approach is to compute the ratio of the average density of its k-nearest neighbors to the density of **x**

$$relative\ density(\mathbf{x}, k) = \frac{\sum_{i=1}^{k} density(\mathbf{y}_i, k)/k}{density(\mathbf{x}, k)}.$$

  - It is possible to replace density(**x**, k) with avg.density(**x**, k) to obtain a more robust measure of relative density
- Similar to the Local Outlier Factor (**LOF**) score

Outliers: Local: LOF

- **Local Outlier Factor (LOF)**
  - LOF is the most well-known and widely used local anomaly detection algorithm
  - It carries the idea of nearest neighbors to determine the anomaly or outlier score
  - In simple words, LOF compares the local density of a point to local density of its k-nearest neighbors and gives a score as final output

https://medium.com/@pramodch/understanding-lof-local-outlier-factor-for-implementation-1f6d4ff13ab9
[Ref.-2]

Outliers: LOF

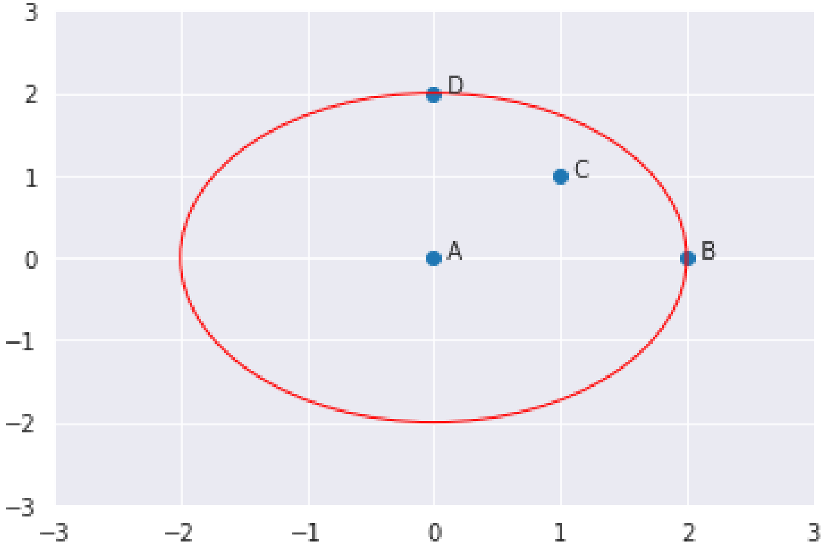
https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- **Local Outlier Factor (LOF)**
  - A point is considered as an outlier based on its local neighborhood (**local outlier**)
  - An outlier is found considering the density of the neighborhood
- **Concepts**
  - K-distance and K-neighbors
  - Reachability distance (RD)
  - Local reachability density (LRD)
  - Local Outlier Factor (LOF)

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- **K-distance** is the distance between the point, and it's K^{th} nearest neighbor
- **K-neighbors** ($N_k(A)$) includes a set of points that lie in or on the circle of radius K-distance



K=2

$||N_2(A)|| = 3$

K-distance of A with K=2

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- **REACHABILITY DENSITY (RD):** defined as the maximum of K-distance of X_j and the distance between X_i and X_j

$$\mathrm{RD}(X_i, X_j) = \max\left(K-\ \text{distance}\ (X_j), \text{distance}\ \ (X_i, X_j)\right)$$

[Ref.-2] In simpler words, it is the distance need to travel from particular point to its neighbor point

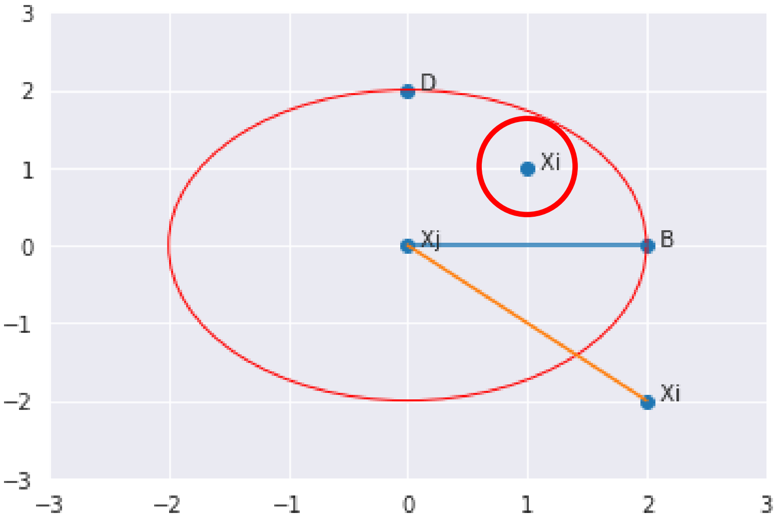


Illustration of reachability distance with K=2

If a point Xi lies within the K-neighbors of Xj, the reachability distance will be K-distance of Xj (blue line), else reachability distance will be the distance between Xi and Xj (orange line)

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

□ LOCAL REACHABILITY DENSITY (LRD)

$$LRD_k(A) = \frac{1}{\sum_{X_j \in N_k(A)} \frac{\mathrm{RD}(A,X_j)}{\|N_k(A)\|}}$$

LRD is inverse of the average reachability distance of A from its neighbors

Intuitively, more the average reachability distance (i.e., neighbors are far from the point), less density of points are present around a particular point. This tells how far a point is from the nearest cluster of points

Low values of LRD implies that the closest cluster is far from the point

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

LOCAL OUTLIER FACTOR (LOF)

$$LOF_k(A) = \frac{\sum_{Xj \in N_k(A)} LRD_k(Xj)}{||N_k(A)||} \times \frac{1}{LRD_k(A)}$$

LRD of each point is used to compare with the average LRD of its K neighbors

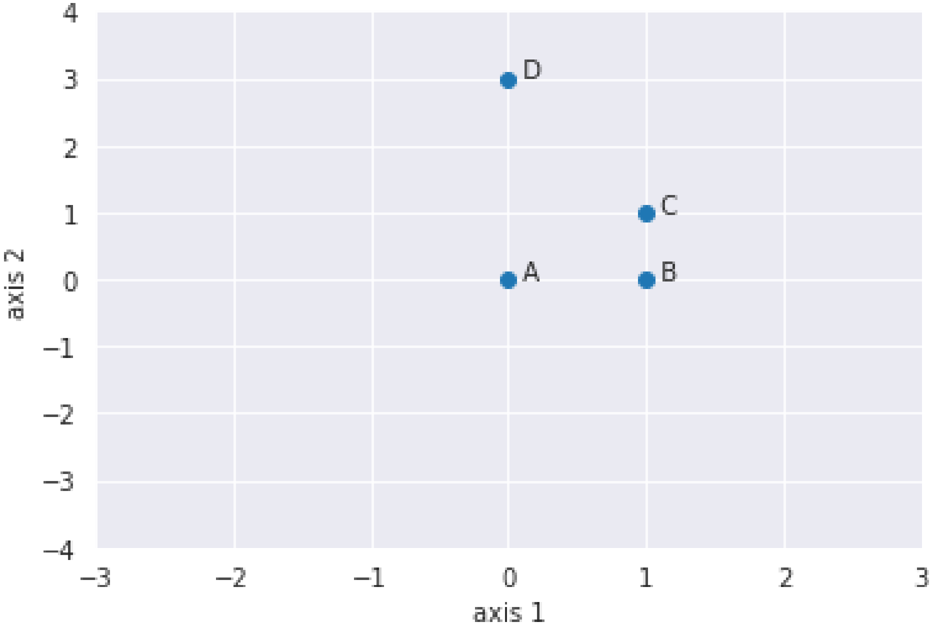
LOF is the ratio of the average LRD of the K neighbors of A to the LRD of A

Intuitively, if the point is not an outlier (inlier), the ratio of average LRD of neighbors is approximately equal to the LRD of a point (because the density of a point and its neighbors are roughly equal). In that case, LOF is nearly equal to 1. On the other hand, if the point is an outlier, the LRD of a point is less than the average LRD of neighbors. Then LOF value will be high (LOF> 1)

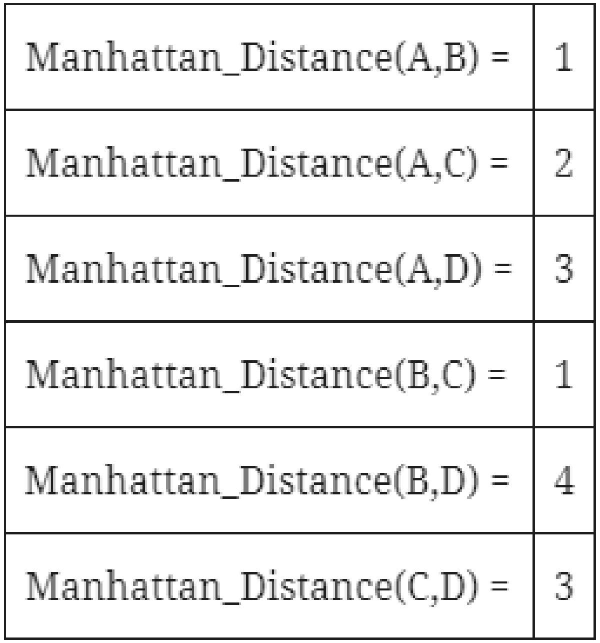
Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- Example



A(0,0), B(1,0), C(1,1), and D(0,3)

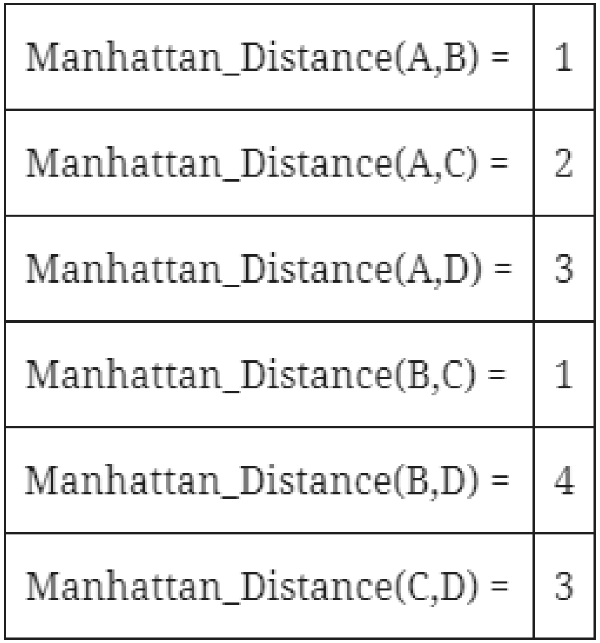


Manhattan_Distance(A,B) =	1
Manhattan_Distance(A,C) =	2
Manhattan_Distance(A,D) =	3
Manhattan_Distance(B,C) =	1
Manhattan_Distance(B,D) =	4
Manhattan_Distance(C,D) =	3

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

□ Example



Manhattan_Distance(A,B) =	1
Manhattan_Distance(A,C) =	2
Manhattan_Distance(A,D) =	3
Manhattan_Distance(B,C) =	1
Manhattan_Distance(B,D) =	4
Manhattan_Distance(C,D) =	3

- K-distance(A) –> since C is the 2ND nearest neighbor of A –> distance(A,C) =2
- K-distance(B) –> since A, C are the 2ND nearest neighbor of B –> distance(B,C) OR distance(B,A) = 1
- K-distance(C) –> since A is the 2ND nearest neighbor of C –> distance(C,A) =2
- K-distance(D) –> since A,C are the 2ND nearest neighbor of D –> distance(D,A) or distance(D,C) =3

- K-neighborhood (A) = {B,C} , $||N_2(A)||$ =2
- K-neighborhood (B) = {A,C}, $||N_2(B)||$ =2
- K-neighborhood (C)= {B,A}, $||N_2(C)||$ =2
- K-neighborhood (D) = {A,C}, $||N_2(D)||$ =2

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- Example

$$LRD_2(A) = \frac{1}{\frac{RD(A,B)+RD(A,C)}{||N_2(A)||}} = \frac{\frac{1}{1+2}}{2} = 0.667$$

$$LRD_2(B) = \frac{1}{\frac{RD(B,A)+RD(B,C)}{||N_2(B)||}} = \frac{\frac{1}{2+2}}{2} = 0.50$$

$$LRD_2(C) = \frac{1}{\frac{RD(C,B)+RD(C,A)}{||N_2(C)||}} = \frac{\frac{1}{1+2}}{2} = 0.667$$

$$LRD_2(D) = \frac{1}{\frac{RD(D,A)+RD(D,C)}{||N_2(D)||}} = \frac{\frac{1}{3+3}}{2} = 0.337$$

RD(A,B) = max(K-distance(B), d(A,B)**) =**
max(1, 1) = 1

RD(A,C) = max(K-distance(C), d(A,C)**) =**
max(2, 2) = 2

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- Example

$$LOF_2(A) = \frac{LRD_2(B) + LRD_2(C)}{\|N_2(A)\|} \times \frac{1}{LRD_2(A)} = \frac{0.5 + 0.667}{2} \times \frac{1}{0.667} = 0.87$$

$$\mathrm{LOF}_2(B) = \frac{LRD_2(A) + LRD_2(C)}{\|N_2(B)\|} \times \frac{1}{LRD_2(B)} = \frac{0.667 + 0.667}{2} \times \frac{1}{0.5} = 1.334$$

$$\mathrm{LOF}_2(C) = \frac{LRD_2(B) + LRD_2(A)}{\|N_2(C)\|} \times \frac{1}{LRD_2(C)} = \frac{0.5 + 0.667}{2} \times \frac{1}{0.667} = 0.87$$

$$\boxed{\mathrm{LOF}_2(D) = \frac{LRD_2(A) + LRD_2(C)}{\|N_2(D)\|} \times \frac{1}{LRD_2(D)} = \frac{0.667 + 0.667}{2} \times \frac{1}{0.337} = 2}$$

Highest LOF among the four points is LOF(D). Therefore, D is an outlier

Outliers: LOF

https://towardsdatascience.com/local-outlier-factor-lof-algorithm-for-outlier-identification-8efb887d9843

- **ADVANTAGES OF LOF**
  - A point will be considered as an outlier if it is at a small distance to the extremely dense cluster. The global approach may not consider that point as an outlier
- **DISADVANTAGES OF LOF**
  - Since LOF is a ratio, it is tough to interpret. There is no specific threshold value above which a point is defined as an outlier. The identification of an outlier is dependent on the problem and the user
  - **[Ref.-2]** In respect to parameter selection, k-value is crucial. Since, LOF is sensitive to chosen k-value
  - **[Ref.-3]** In higher dimensions, the LOF algorithm detection accuracy gets effected https://www.geeksforgeeks.org/local-outlier-factor/

Strengths/Weaknesses of Distance-based Approaches

- They are non-parametric
- Can be used where a reasonable proximity measure can be defined between instances
- They are quite intuitive and visually appealing when the data can be displayed in two- or three-dimensional scatter plots
- Expensive: $O(n^2)$
- Sensitive to parameters: distance, k, n, d, ...
- Sensitive to variations in density
- Distance becomes less meaningful in high-dimensional space

Clustering-based Approaches

- Use clusters to represent the normal class
  - Assumption: normal instances appear close to each other and hence can be grouped into clusters
    - Anomalies are identified as instances that do not fit well in the clustering or appear in small clusters that are far apart from the clusters of the normal class
      - Methods can be categorized into two types:
        - Methods that consider small clusters as anomalies
        - Methods that define a point as anomalous if does not fit the clustering well, typically as measured by distance from a cluster center

1. Finding Anomalous Clusters

- This approach assumes the presence of clustered anomalies in the data, where the anomalies appear in tight groups of small size
  - Clustered anomalies appear when the anomalies are being generated from the same anomalous class

Finding Anomalous Clusters

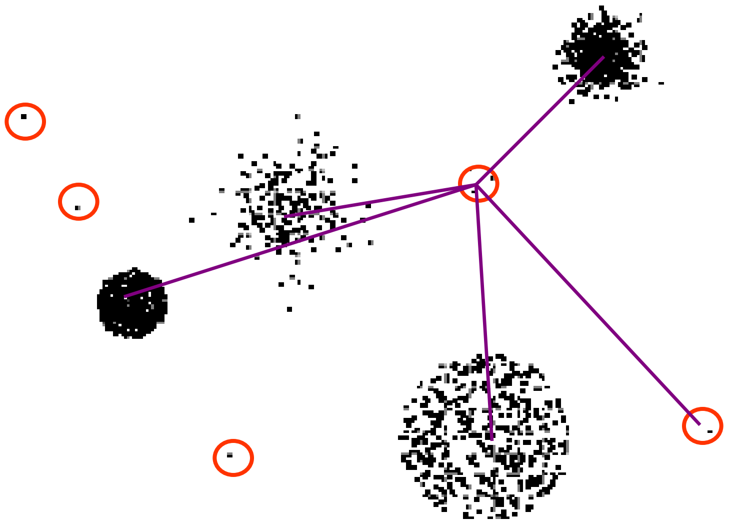
- A basic approach is to cluster the overall data and flag clusters that are either too small in size or too far from other clusters
  - Every cluster can be represented by its prototype, e.g., the centroid of the cluster
  - Treat every prototype as a point and straightforwardly identify clusters that are distant from the rest

2. Finding Anomalous Instances

- Another way of describing an anomaly is as an instance that cannot be explained well by any of the normal clusters
- A basic approach is to first cluster all the data (comprised mainly of normal instances) and then assess the degree to which every instance belongs to its respective cluster
  - Instances that are quite distant from their respective cluster centroids can thus be identified as anomalies

Finding Anomalous Instances

- An object is a cluster-based outlier if it does not strongly belong to any cluster
  - For prototype-based clusters, an object is an outlier if it is not close enough to a cluster center
    - Outliers can impact the clustering produced
  - For density-based clusters, an object is an outlier if its density is too low
    - Can't distinguish between noise and outliers
  - For graph-based clusters, an object is an outlier if it is not well connected



Strengths/Weaknesses of Clustering-Based Approaches

- Many clustering techniques can be used
- Can be difficult to decide on a clustering technique
- Can be difficult to decide on number of clusters
- Outliers can distort the clusters

Reconstruction-based Approaches

- Based on assumptions there are patterns in the distribution of the normal class that can be captured using lower-dimensional representations
- Reduce data to lower dimensional data
  - Principal Components Analysis (PCA) or Autoencoders, for example
- Measure the reconstruction error for each object
  - The difference between original and reduced dimensionality version

Reconstruction Error

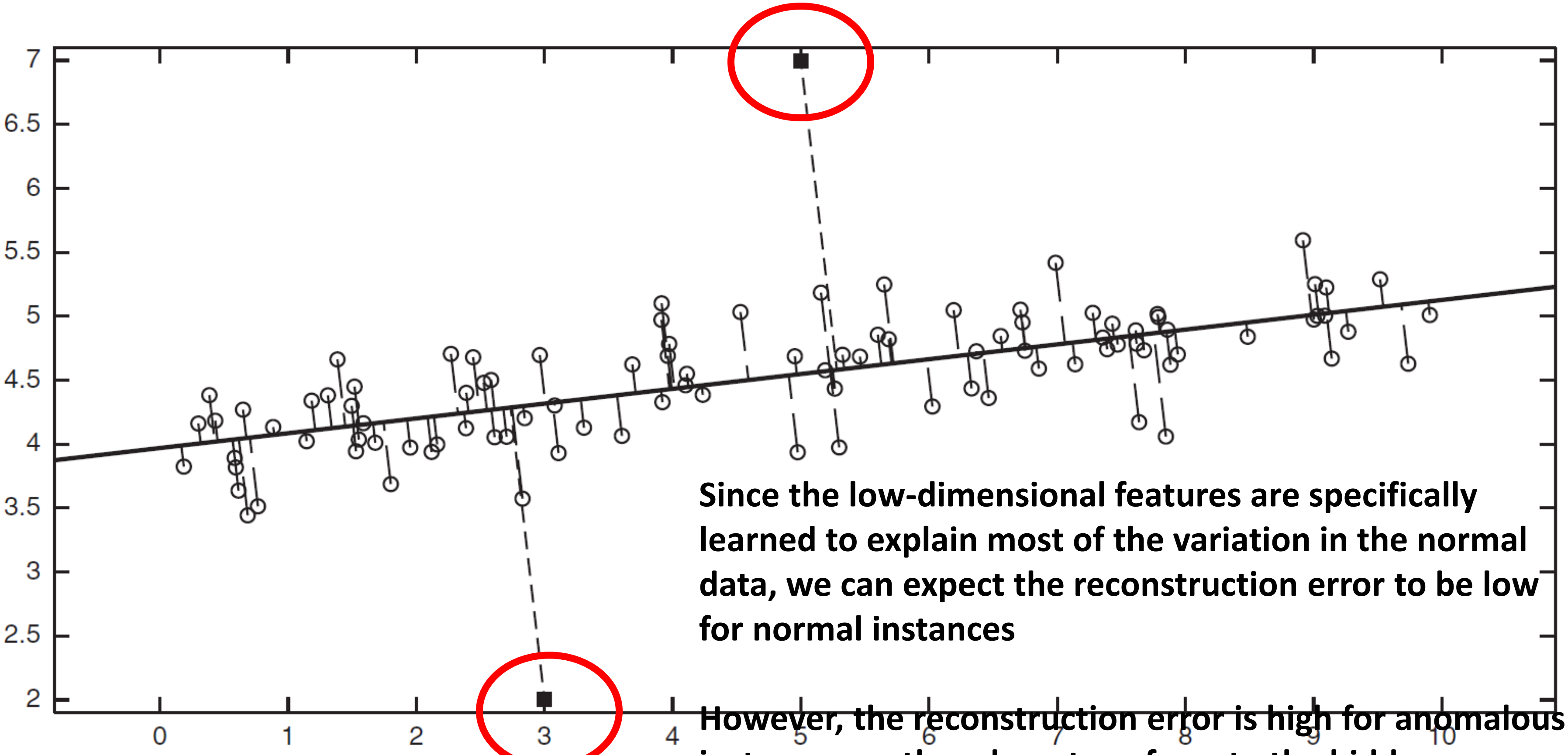
- Let $\mathbf{x}$ be the original data object
- Find the representation of the object in a lower dimensional space

- Project the object back to the original space
  - Call this object $\hat{\mathbf{x}}$

$$\text{Reconstruction Error}(\mathbf{x}) = ||\mathbf{x} - \hat{\mathbf{x}}||^2$$

- Objects with large reconstruction errors are anomalies

Reconstruction of two-dimensional data



Since the low-dimensional features are specifically learned to explain most of the variation in the normal data, we can expect the reconstruction error to be low for normal instances

However, the reconstruction error is high for anomalous instances, as they do not conform to the hidden structure of the normal class

Strengths and Weaknesses

- Does not require assumptions about distribution of normal class

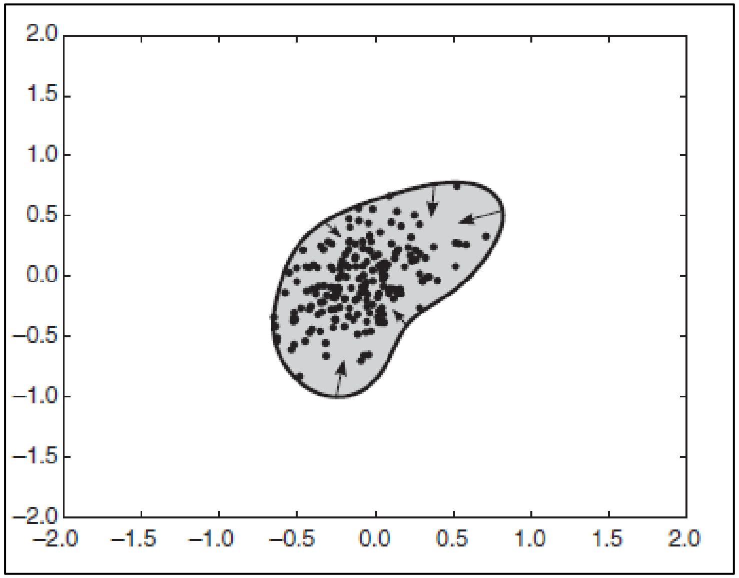
- Can use many dimensionality reduction approaches

- The reconstruction error is computed in the original space
  - This can be a problem if dimensionality is high

One-class Classification

- Learn a decision boundary in the attribute space that encloses all normal objects on one side of the boundary
- This is in contrast to binary classification approaches that learn boundaries to separate objects from two classes

The decision boundary of a one-class classification problem attempts to enclose the normal instances on the same side of the boundary

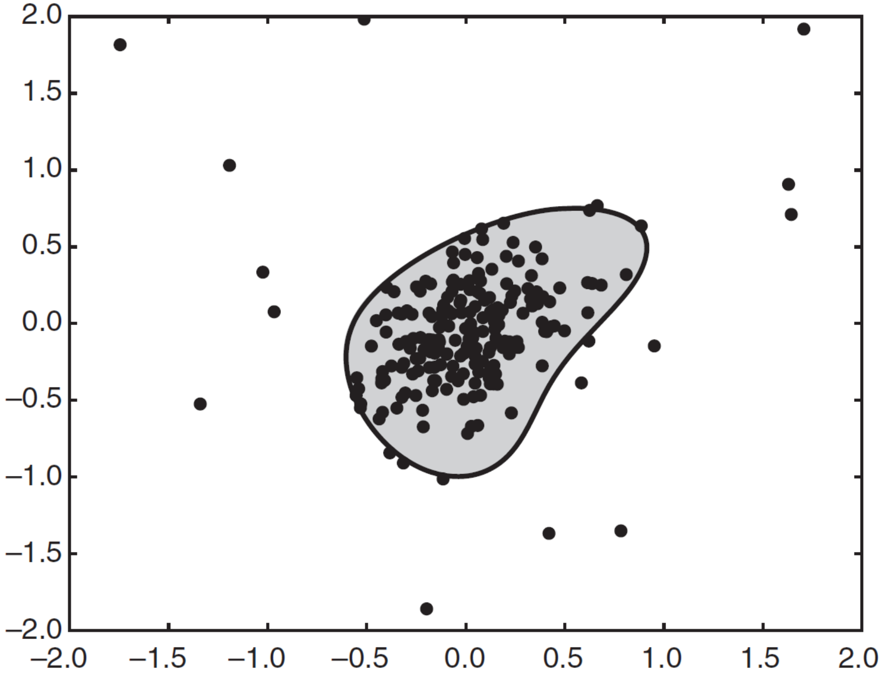


One-class Classification

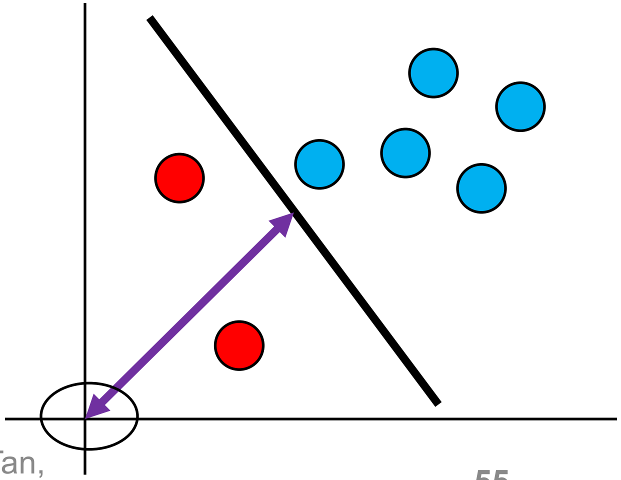
- Instead of learning the distribution of the normal class, the focus is on modeling the boundary of the normal class
  - Learning the boundary is indeed what we need to distinguish anomalies from normal objects
- **One-class SVM**
  - Uses the training instances from the normal class to learn its decision boundary

Finding Outliers with a One-Class SVM

- Decision boundary with $\nu = 0.1$ $(Tr = 200, 20)$

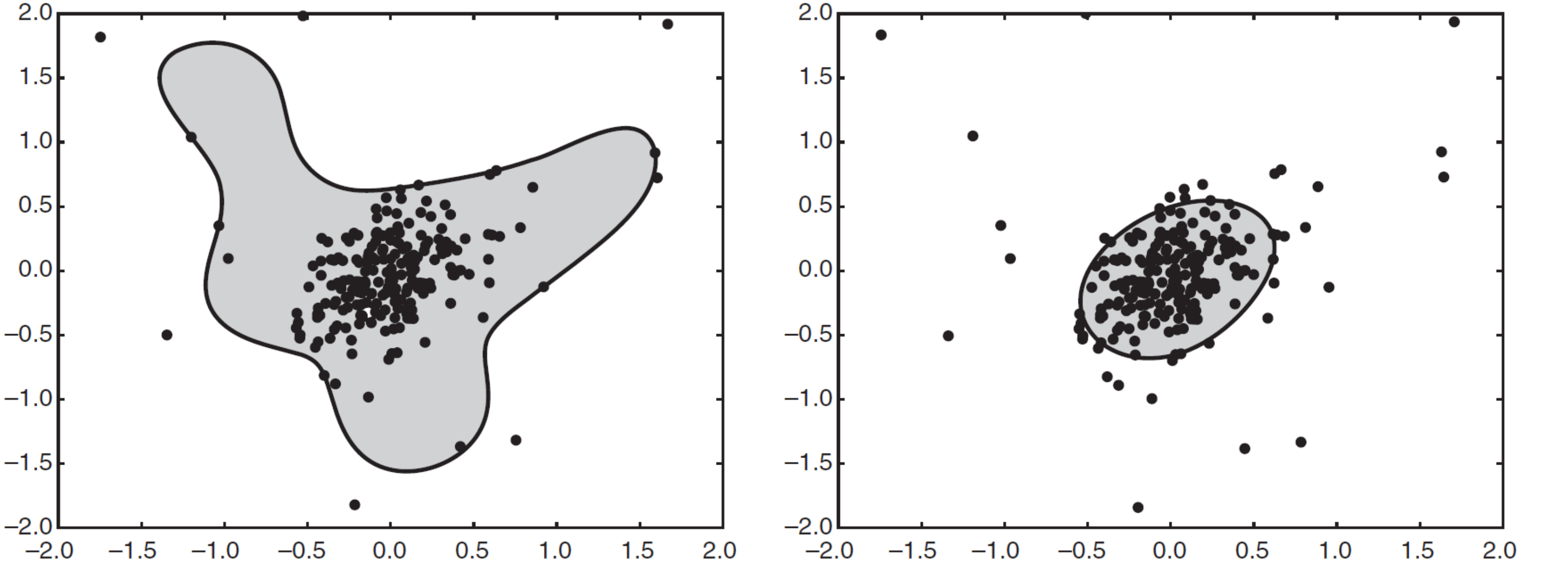


ν = represents an upper bound on the fraction of training instances that can be tolerated as anomalies while learning the hyperplane



Finding Outliers with a One-Class SVM

- Decision boundary with $\nu = 0.05$ (10) and $\nu = 0.2$ (40)



(a) $\nu = 0.05$.

(b) $\nu = 0.2$.

Strengths and Weaknesses

- Strong theoretical foundation

- Choice of ν is difficult

- Computationally expensive

Information Theoretic Approaches

- Assume that the normal class can be represented using compact representations, also known as codes
- Instead of explicitly learning such representations, the focus is to quantify the amount of information required for encoding them
- There are a number of approaches for quantifying the information content (also referred to as complexity) of a data set
  - For categorical variable, for example, we can assess its information content using the entropy measure

Information Theoretic Approaches

- Basic information theoretic approach for anomaly detection
  - Key idea is to measure how much information decreases when you delete an observation

$$Gain(x) = Info(D) - Info(D \setminus x)$$

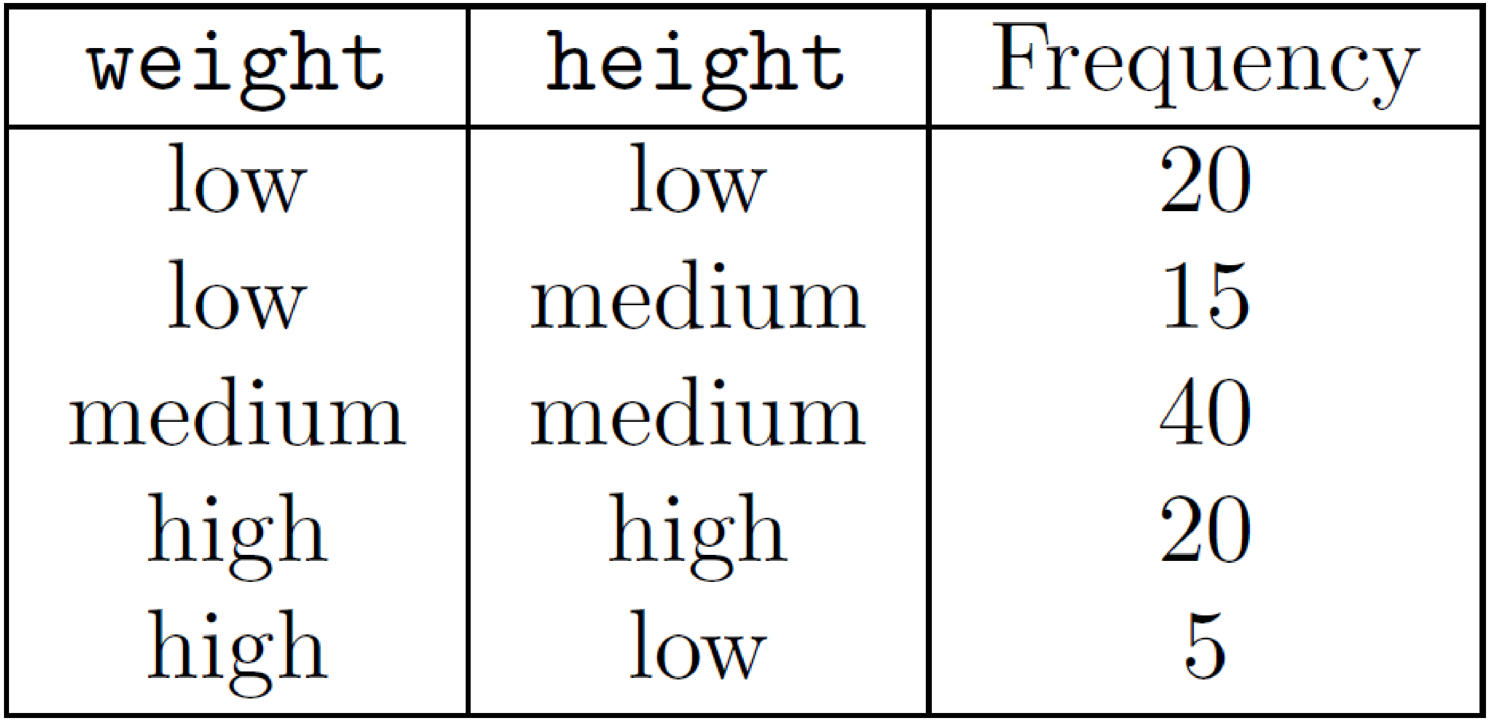
  - Anomalies should show higher gain
    - ◆ Anomalies are expected to be surprising, and thus, their elimination should result in a substantial reduction in the information content
  - Normal points should have less gain
  - Gain(x) = measure of anomaly score

Information Theoretic Approaches

- Typically, the reduction in information content is measured by eliminating a subset of instances (that are deemed anomalous) and not just a single instance
  - This is because most measures of information content are not sensitive to the elimination of a single instance
  - It is thus necessary to identify the smallest subset of instances X that show the largest value of Gain(X) upon elimination
    - This is a non-trivial problem requiring exponential time complexity

Information Theoretic Example

- Survey of height and weight for 100 participants



weight	height	Frequency
low	low	20
low	medium	15
medium	medium	40
high	high	20
high	low	5

- Eliminating last group give a gain of $2.08 - 1.89 = 0.19$

Strengths and Weaknesses

- Solid theoretical foundation

- Theoretically applicable to all kinds of data

- Heavily depends on the choice of the measure used for capturing the information content of data set

- Difficult and computationally expensive to implement in practice

Evaluation of Anomaly Detection

- If class labels are present, then use standard evaluation approaches for rare class such as precision, recall, false positive rate (false alarm rate), false negative rate (miss rate)

$$FP\ Rate = \alpha = \frac{FP}{TN + FP}$$

$$FN\ Rate = \beta = \frac{FN}{FN + TP}$$

TP = min
TN = maj

high values = the users will turn off the system since it is more distracting than useful

high values = miss a lot of crucial anomalies and you will lose trust in the system

- For unsupervised anomaly detection use measures provided by the anomaly method
  - Reconstruction error or gain, for example

Evaluation of Anomaly Detection

- Can also look at histograms of anomaly scores
  - The techniques discussed assume that only a relatively small fraction of the data consists of anomalies
  - The majority of anomaly scores should be relatively low, with a smaller fraction of scores toward the high end
    - This assumes that a higher score indicates an instance is more anomalous
  - Thus, by looking at the distribution of the scores via a histogram, we can assess whether the approach we are using generates scores that behave in a reasonable manner

Distribution of Anomaly Scores

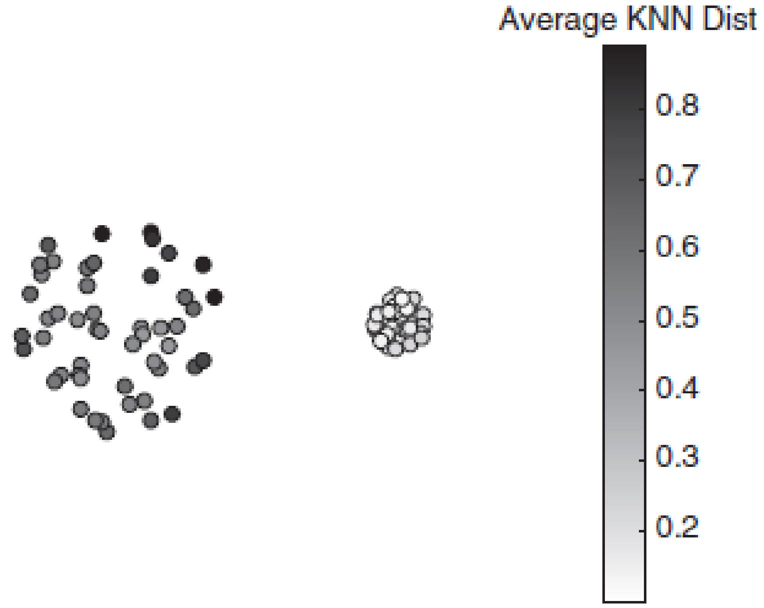


Figure 9.17. Anomaly score based on average distance to fifth nearest neighbor.

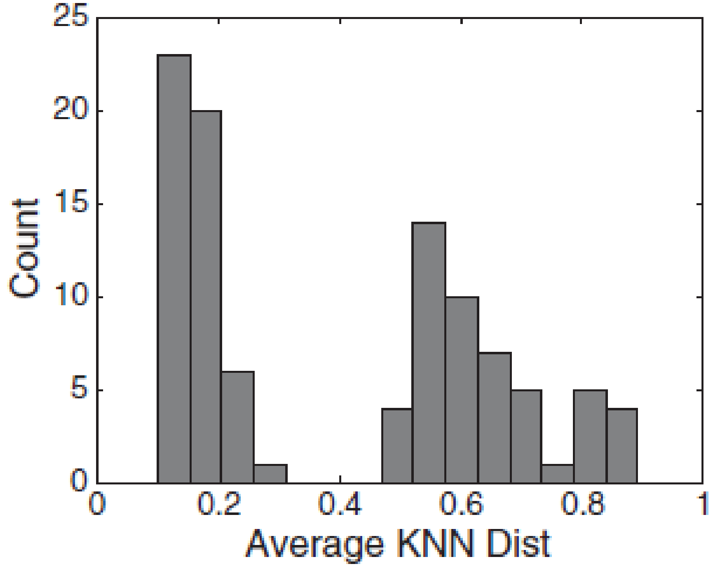


Figure 9.19. Histogram of anomaly score based on average distance to the fifth nearest neighbor.

Distribution of Anomaly Scores

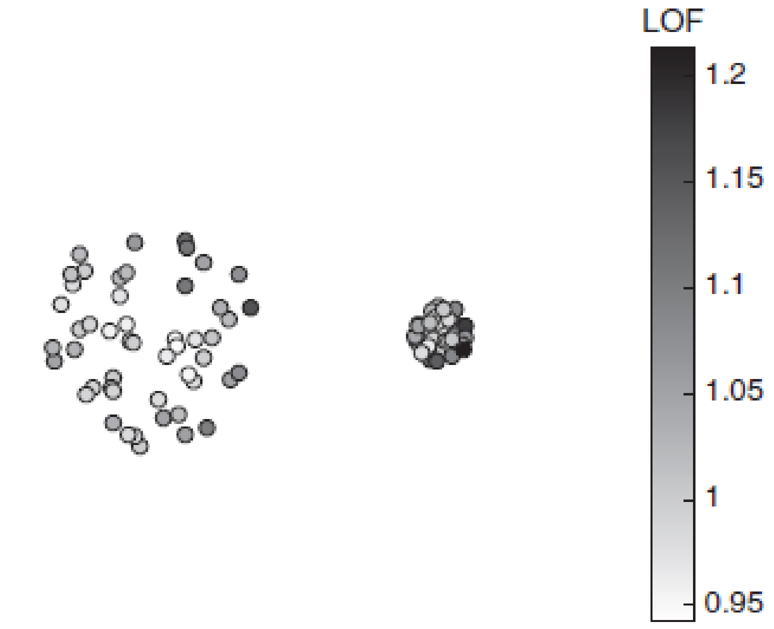


Figure 9.18. Anomaly score based on LOF using five nearest neighbors.

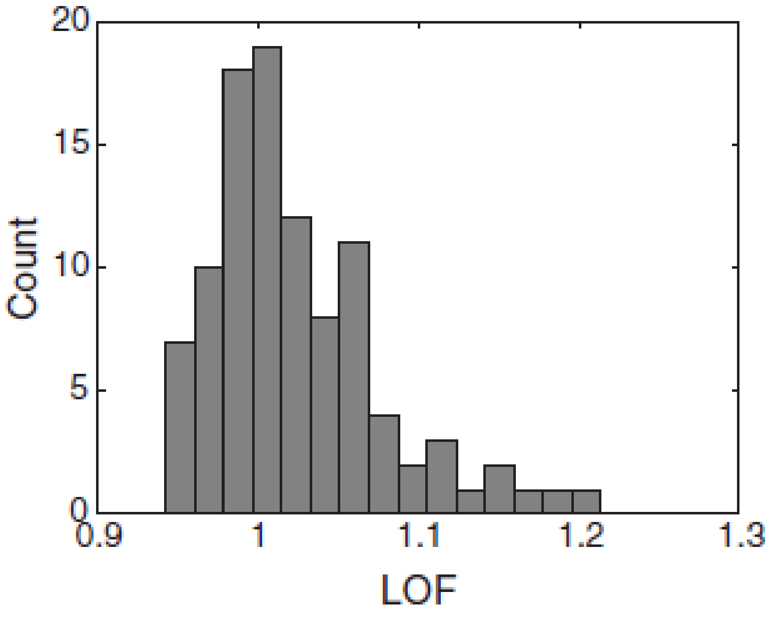


Figure 9.20. Histogram of LOF anomaly score using five nearest neighbors.

The distribution of anomaly scores should look similar to that of the LOF scores in this example

Recursos

- scikit-learn
  - Novelty and Outlier Detection [https://scikit-learn.org/stable/modules/outlier_detection.html]
    - One-Class SVM
    - Isolation Forest
    - Local Outlier Factor (LOF)
  - Outlier detection
    - The training data contains outliers which are defined as observations that are far from the others
  - Novelty detection
    - The training data is not polluted by outliers and we are interested in detecting whether a new observation is an outlier

Isolation Forest

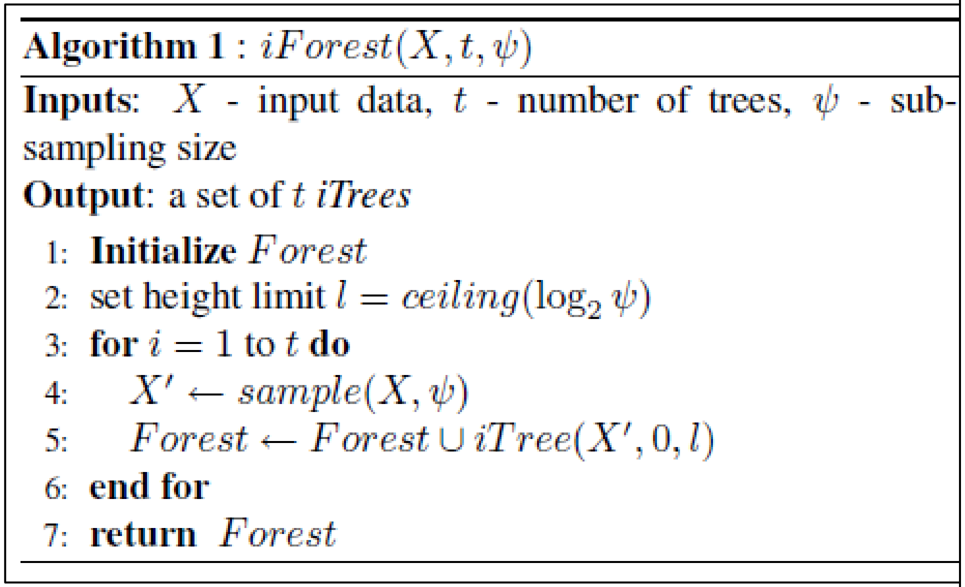
https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/

- Similar to Random Forests
  - However, since there are no pre-defined labels here, it is an unsupervised model
- Ensemble of binary decision trees
  - Samples that travel deeper into the tree are less likely to be anomalies as they required more cuts to isolate them
  - Samples which end up in shorter branches indicate anomalies as it was easier for the tree to separate them from other observations

Isolation Forest

https://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf

- Ensemble of binary decision trees
  - Each tree in an Isolation Forest is called an Isolation Tree (iTree)

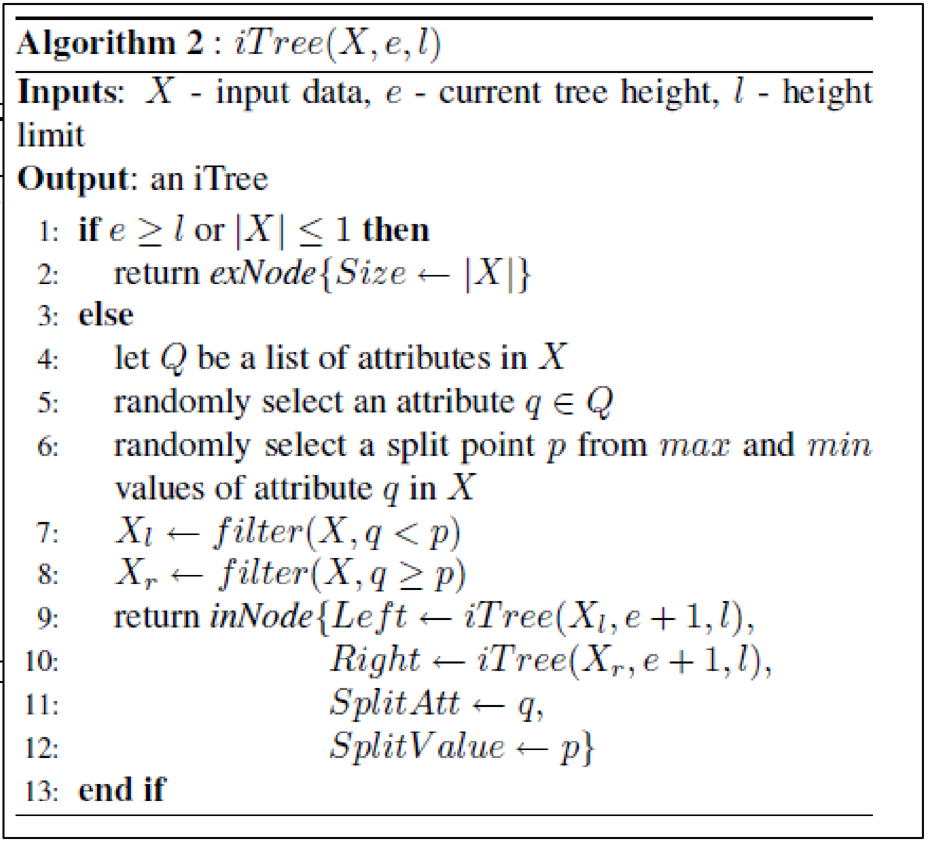


Algorithm 1 : $iForest(X, t, \psi)$

Inputs: X - input data, t - number of trees, ψ - sub-sampling size

Output: a set of t *iTrees*

```
Initialize $Forest$
set height limit $l = ceiling(\log_2 \psi)$
for $i = 1$ to $t$ do
    $X' \leftarrow sample(X, \psi)$
    $Forest \leftarrow Forest \cup iTree(X', 0, l)$
end for
return $Forest$
```



Algorithm 2 : $iTree(X, e, l)$

Inputs: X - input data, e - current tree height, l - height limit

Output: an iTree

```
if $e \geq l$ or $|X| \leq 1$ then
    return exNode{$Size \leftarrow |X|$}
else
    let $Q$ be a list of attributes in $X$
    randomly select an attribute $q \in Q$
    randomly select a split point $p$ from $max$ and $min$
    values of attribute $q$ in $X$
    $X_l \leftarrow filter(X, q < p)$
    $X_r \leftarrow filter(X, q \geq p)$
    return inNode{$Left \leftarrow iTree(X_l, e + 1, l)$,
                  $Right \leftarrow iTree(X_r, e + 1, l)$,
                  $SplitAtt \leftarrow q$,
                  $SplitValue \leftarrow p$}
end if
```

Proximity-based (Distance-based)

Isolation Forest

https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/

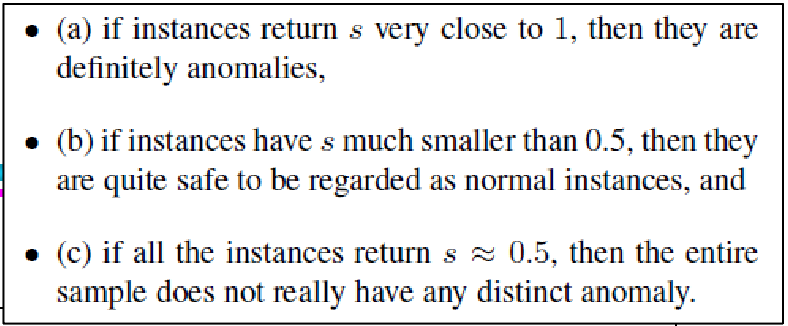
- Algorithm
  - A random sub-sample of the data is selected
  - Branching
    - Branching of the tree starts by selecting a random feature (from the set of all N features)
    - After, branching is done on a random threshold (any value in the range of minimum and maximum values of the selected feature)
    - If the value of a data point is less than the selected threshold, it goes to the left branch else to the right – a node is split into left and right branches
    - The process is continued recursively till each data point is completely isolated or till max depth (if defined) is reached
  - The above steps are repeated

Isolation Forest

https://www.analyticsvidhya.com/blog/2021/07/anomaly-detection-using-isolation-forest-a-complete-guide/

- During scoring, a data point is traversed through all the trees which were trained earlier
- An anomaly score is assigned to each of the data points based on the depth of the tree required to arrive at that point – an aggregation of the depth obtained from each of the iTrees

Isolation Forest



- (a) if instances return s very close to 1, then they are definitely anomalies,
- (b) if instances have s much smaller than 0.5, then they are quite safe to be regarded as normal instances, and
- (c) if all the instances return $s \approx 0.5$, then the entire sample does not really have any distinct anomaly.

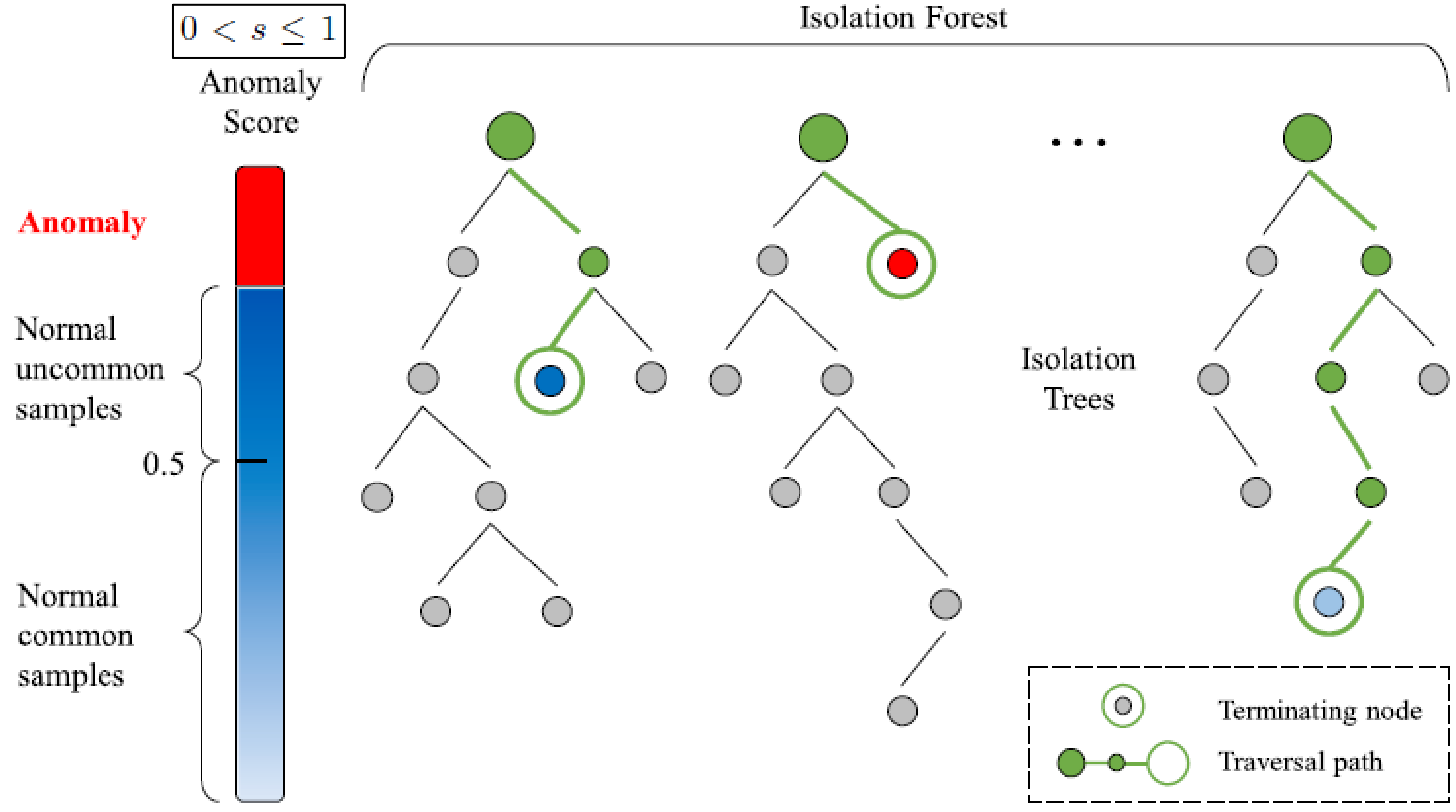


Fig. 3. Anomaly detection with *i*Forest.

https://www.sciencedirect.com/science/article/abs/pii/S1474034620301105?via%3Dihub

Recursos

- https://pyod.readthedocs.io/en/latest/
  - ADBench: Anomaly Detection Benchmark [https://arxiv.org/pdf/2206.09426.pdf]