Topic 3

Outlier Detection

[Jiawei Han, Micheline Kamber, Jian Pei. 2011. Data Mining Concepts and Techniques. 3rd Ed. Morgan Kaufmann. ISBN: 9380931913.]
1. Classification-Based Approaches  
(Supervised, one-class model - SVMs)

2. Clustering-Based Approaches  
(Unsupervised, CBLOF)

3. Proximity-Based Approaches  
(Distance-Based: \( DB(r, \pi) \), Density-Based: \( LOF_k(o) \))

4. Statistical Approaches  
(Data \( \notin [\mu - \sigma, \mu + \sigma] \) is outlier, Grubbs’ Test, Mahalanobis distance)
• **Outlier detection** (a.k.a. **anomaly detection**) is the process of finding data objects with behaviors that are very different from expectation. Such objects are called **outliers** or **anomalies**.

• Outlier detection can be used for intrusion detection, credit card fraud detection, medical diagnosis, fault detection, law enforcement, earth science, anomalous activity in social networks, and so on.
Categories of Outlier Detection Methods

• Outlier detection methods can be categorized as follows.
  - **Statistical** methods,
  - **Proximity-based** methods,
  - **Classification-based** methods,
  - **Clustering-based** methods.
Statistical Methods

- **Statistical** methods (also known as **model-based methods**) make assumptions of data normality. They assume that normal data objects are generated by a statistical (stochastic) model, and that data not following the model are outliers.
Statistical Methods

- **Statistical** methods: use data distribution (e.g., Gaussian aka normal).
- Two main types of statistical methods are **parametric** and **nonparametric** (e.g., histogram) methods. Parametric methods consider **univariate** data (involving one attribute/variable) and **multivariate** data (involving two or more attributes or variables, Mahalanobis distance, $\chi^2$-statistic can be used).
Proximity-Based Methods

- **Proximity-based** methods assume that an object is an outlier if the nearest neighbors of the object are far away in feature space. That is, the proximity of the object to its neighbors significantly deviates from the proximity of most of the other objects to their neighbors in the same data set.
Proximity-Based Methods

- **Proximity-based** methods: use $k$-nearest neighbors ($k$-NN) or $\varepsilon$-neighborhood. Two major types of proximity-based methods are **distance-based** and **density-based** methods. A distance-based method can detect **global outliers** only. A density-based method can identify **local outliers**.
Classification-Based Methods

- **Classification-based methods** (supervised): build one-class model using **support vector machines** (SVMs).

  (semi-supervised method is a combination of supervised and unsupervised methods)
Clustering-Based Methods

• Clustering-based methods assume that the normal data objects belong to large and dense clusters, whereas outliers belong to small or sparse clusters, or do not belong to any clusters.

• Clustering-based methods (unsupervised): examines the relationship between objects and clusters. An outlier is an object that does not belong to any cluster, belongs to a small (or sparse) cluster, or belongs to a remote cluster.
**Example:** Most objects follow a roughly Gaussian distribution. However, the objects in region $R$ are significantly different. It is unlikely that they follow the same distribution as the other objects in the data set. Thus, the objects in $R$ are outliers in the data set.
Categories of Outliers

• Outliers can be classified into three categories, namely global outliers, contextual (or conditional) outliers, and collective outliers.
• A data object is a global outlier if it deviates significantly from the rest of the data set.
• Most outlier detection methods are aimed at finding global outliers.
**Example:** The points in region $R$ significantly deviate from the rest of the data set, and hence are examples of **global outliers** (distance-based method). $o_3$ is a global outlier.
Contextual Outliers

• The temperature today is 28°C. Is it an outlier? It depends on the time and location!
  - If it is in winter in Toronto, yes, it is an outlier.
  - If it is a summer day in Toronto, then it is normal.

• In a given data set, a data object is a contextual outlier (aka conditional outliers) if it deviates significantly with respect to (w.r.t.) a specific context of the object.
In contextual outlier detection, the attributes of the data objects in question are divided into two groups:

- **Contextual attributes**: The contextual attributes of a data object define the object’s context. In the temperature example, the contextual attributes may be date and location.
- **Behavioral attributes**: These define the object’s characteristics, and are used to evaluate whether the object is an outlier in the context to which it belongs. In the temperature example, the behavioral attributes may be the temperature, humidity, and pressure.
Local Outliers

- An object in a data set is a **local outlier** if its density significantly deviates from the local area in which it occurs. \( o_1 \) and \( o_2 \) are local outliers w.r.t. cluster \( C_1 \) (density-based method).
Collective Outliers

- Given a data set, a subset of data objects forms a **collective outlier** if the objects as a whole deviate significantly from the entire data set. Importantly, the individual data objects may not be outliers.
Example: The black objects as a whole form a collective outlier because the density of those objects is much higher than the rest in the data set. However, every black object individually is not an outlier w.r.t. the whole data set.
Global, Contextual, Collective Outliers

- In summary,
  - Global outlier detection is the simplest.
  - Contextual outlier detection requires background information to determine contextual attributes and contexts.
  - Collective outlier detection requires background information to model the relationship among objects to find groups of outliers.
Categories of Outlier Detection Methods

• Outlier detection methods can be categorized as statistical methods, proximity-based methods, classification-based methods, and clustering-based methods.

• Outlier detection models can also be divided into supervised methods (classification), unsupervised methods (clustering), and semi-supervised methods (combination of classification and clustering).
1. Classification-Based Approaches

- Outlier detection can be treated as a classification problem if a training data set with class labels is available.
- The general idea of classification-based outlier detection (classification-BOD) methods is to train a classification model that can distinguish normal data from outliers.
1. Classification-Based Approaches

- Consider a training set that contains samples labeled as “normal” and others labeled as “outlier.” A classifier can then be constructed based on the training set.

- Classification-BOD methods often use a one-class model. That is, a classifier is built to describe only the normal class. Any samples that do not belong to the normal class are regarded as outliers.
Example: Outlier detection using a one-class model. Consider the training set where white points are samples labeled as “normal” and black points are samples labeled as “outlier.”
1. Classification-Based Approaches

• To build a model for outlier detection, we can learn the decision boundary of the normal class using classification methods such as support vector machines (SVMs).

• Given a new object, if the object is within the decision boundary of the normal class, it is treated as a normal case. If the object is outside the decision boundary, it is declared an outlier.
1. Classification-Based Approaches

- Classification-based methods and clustering-based methods can be combined to detect outliers in a semi-supervised learning way.
Example: Outlier detection by semi-supervised learning. Consider objects are labeled as either “normal” or “outlier,” or have no label at all.

- Using a clustering-based approach, we find a large cluster $C$, and a small cluster $C_1$. Because some objects in $C$ carry the label “normal,” we can treat all objects in this cluster (including those without labels) as normal objects.
1. Classification-Based Approaches

- We use the one-class model of this cluster $C$ to identify normal objects in outlier detection.
- Similarly, because some objects in cluster $C_1$ carry the label "outlier," we declare all objects in $C_1$ as outliers.
- Any object that does not fall into the model for $C$ (e.g., $a$) is considered an outlier as well.
1. Classification-Based Approaches
(Supervised, one-class model - SVMs)

2. Clustering-Based Approaches
(Unsupervised, CBLOF)

3. Proximity-Based Approaches
(Distance-Based: $DB(r, \pi)$, Density-Based: $LOF_k(o)$)

4. Statistical Approaches
(Data $\notin [\mu - \sigma, \mu + \sigma]$ is outlier, Grubbs’ Test, Mahalanobis distance)
2. Clustering-Based Approaches

• Clustering-based approaches detect outliers by examining the relationship between objects and clusters.
• An outlier is an object that
  - does not belong to any cluster,
  - is far from its closest cluster, or
  - belongs to a small (or sparse) cluster.
2. Clustering-Based Approaches

- There are three approaches to clustering-based outlier detection (clustering-BOD).
  1. Does object belong to any cluster? If not, then it is identified as outlier.
  2. Consider distance between an object and the cluster to which it is closest. If distance is large, the object is likely outlier w.r.t. the cluster. This approach detects individual outliers w.r.t clusters.
  3. Is the object part of a small or sparse cluster? If yes, then all the objects in that cluster are outliers.
2. Clustering-Based Approaches

**Example:** Detect outliers as objects that do not belong to any cluster.

- Data points were clustered into two clusters by using density-based clustering method (e.g., DBSCAN), black points belong to clusters. White point $a$ does not belong to any cluster so it is an outlier.
2. Clustering-Based Approaches

**Example:** Detect an outlier using distance from the object to its closest cluster.

- Using *k*-means clustering method, data points were clustered into three clusters, as shown below using different symbols ○, ●, ⊕. The center of each cluster is marked with a + sign.
2. Clustering-Based Approaches

For each object \( o \), we can assign an outlier score to the object according to the distance between the object and the center that is closest to the object.

Suppose the closest center to \( o \) is \( c_o \).

The distance between \( o \) and \( c_o \) is \( \text{dist}(o, c_o) \), and the average distance between \( c_o \) and the objects \( o' \) in \( c_o \) is \( \text{dist}_a(o', c_o) \). The ratio \( \text{dist}(o, c_o)/\text{dist}_a(o', c_o) \) measures how \( \text{dist}(o, c_o) \) stands out from the average. The larger the ratio, the farther away \( o \) is relative from the center, and the more likely \( o \) is an outlier.
2. Clustering-Based Approaches

- Outliers \( a, b, \) and \( c \) are far away from the clusters to which they are closest (w.r.t. the cluster centers).
2. Clustering-Based Approaches

3. Is the object part of a small or sparse cluster? If yes, then all the objects in that cluster are outliers.

• Some outliers may be similar and form a small cluster (e.g., in intrusion detection, hackers who use similar tactics to attack a system may form a cluster).
2. Clustering-Based Approaches

**Example**: Detect outliers in small clusters. The data points form three clusters: large clusters $C_1$ and $C_2$, and a small cluster $C_3$. Object $o$ does not belong to any cluster.

- Using CBLOF, $FindCBLOF$ can identify $o$ and points in cluster $C_3$ as outliers.
2. Clustering-Based Approaches

• An approach to clustering-BOD identifies small or sparse clusters and declares the objects in those clusters to be outliers.

• An example of this approach is the **FindCBLOF (cluster-based local outlier factor)** algorithm.
2. Clustering-Based Approaches

- The **FindCBLOF** algorithm works as follows.

1. Find clusters in a data set, and sort them according to decreasing size. The algorithm assumes that most of the data points are not outliers. It uses a parameter $\alpha$ ($0 \leq \alpha \leq 1$) to distinguish large from small clusters. Any cluster that contains at least a percentage $\alpha$ (e.g., $\alpha = 90\%$) of the data set is considered a large cluster. The remaining clusters are referred to as small clusters.
2. Clustering-Based Approaches

2. To each data point, assign a **cluster-based local outlier factor** (CBLOF).
   - For a point belonging to a large cluster, its CBLOF is the product of the cluster’s size and the distance between the point and the cluster.
   - For a point belonging to a small cluster, its CBLOF is calculated as the product of the size of the small cluster and the distance between the point and the closest large cluster.
2. Clustering-Based Approaches

- Data set \( D \) is clustered into \( k \) clusters \( C_1, C_2, \ldots, C_b, C_{b+1}, \ldots, C_k \), where
- \( |C_1| \geq |C_2| \geq \ldots \geq |C_k| \).
- \( b \) is the boundary of large clusters (LC) and small clusters (SC),
- \( |C_1| \geq |C_2| \geq \ldots \geq |C_b| \geq \alpha \times |D| \), and
- \( |C_b|/|C_{b+1}| \geq \beta \) (e.g., \( \alpha = 90\% \), \( \beta = 4 \) or 5).
2. Clustering-Based Approaches

• For each object \( o \in D \), CBLOF(\( o \)) is defined as

\[
CBLOF(\( o \)) = \begin{cases} 
|C_i| \times \min \{dist(\( o \), \( C_j \))\} \text{ where } \\
o \in C_i, C_i \in SC, C_j \in LC, j = 1..b \\
|C_i| \times dist(\( o \), \( C_i \)) \text{ where } o \in C_i, C_i \in LC
\end{cases}
\]
2. Clustering-Based Approaches

• The larger the CBLOF value, the more similar (i.e., small distance) the point and the cluster are.
• The CBLOF score can detect outlier points that are far from any clusters.
• Small clusters that are far from any large cluster are considered to consist of outliers.
• The points with the lowest CBLOF scores are suspected outliers.
Example: Detect outliers in small clusters. The data points form three clusters: large clusters, $C_1$ and $C_2$, and a small cluster, $C_3$. Object $o$ does not belong to any cluster.
2. Clustering-Based Approaches

- Using CBLOF, *FindCBLOF* can identify *o* and the points in cluster $C_3$ as outliers.
- For *o*, the closest large cluster is $C_1$. The CBLOF(*o*) is simply the similarity between *o* and $C_1$, which is small (i.e., large distance).
- For points in $C_3$, closest large cluster is $C_2$. There are three points in small cluster $C_3$ (i.e., $|C_3| = 3$), similarity between those points and cluster $C_2$ is low (i.e., large distance). Thus, CBLOF scores of points in $C_3$ are small.
2. Clustering-Based Approaches

• Clustering-based approaches may incur high computational costs if they have to find clusters before detecting outliers.
• Clustering-BOD methods have the following advantages.
  - They can detect outliers without requiring any labeled data (i.e., unsupervised).
  - They work for many data types.
• Weakness of clustering-BOD is its effectiveness, which depends highly on clustering method used.
1. Classification-Based Approaches
(Supervised, one-class model - SVMs)

2. Clustering-Based Approaches
(Unsupervised, CBLOF)

3. Proximity-Based Approaches
(Distance-Based: $DB(r, \pi)$, Density-Based: $LOF_k(o)$)

4. Statistical Approaches
(Data $\notin [\mu - \sigma, \mu + \sigma]$ is outlier, Grubbs’ Test, Mahalanobis distance)
3. Proximity-Based Approaches

- Two main types of proximity-based outlier detection methods are **distance-based** and **density-based** methods.
- A **distance-based** outlier detection (distance-BOD) method consults the neighborhood of an object, which is defined by a given radius $r$ (or $\varepsilon$).
- An object is considered as an outlier if its neighborhood does not have enough other points.
• A **density-based** outlier detection (density-BOD) method investigates the density of an object and that of its neighbors.

- An object is identified as an outlier if its density is relatively much lower than that of its neighbors.
Distance-Based Outlier Detection

• For each object $o$ in given data set $D$, we can examine the number of other objects in the $r$-neighborhood of $o$, where $r$ (or $\varepsilon$) is a distance threshold defined by a user.
• If most of the objects in $D$ are far from $o$ (i.e., they are not in the $r$-neighborhood of $o$), then $o$ can be regarded as an outlier.
**DB(r, π)-Outlier Detection Method**

- Let $r$ ($r \geq 0$) be a distance threshold and $\pi$ ($0 < \pi \leq 1$) be a fraction threshold (i.e., percentage, e.g., $\pi = 0.15$).

- An object $o$ is a $DB(r, \pi)$-outlier if
  $$|\{o' \mid dist(o, o') \leq r, o' \neq o\}| \leq \lfloor \pi \cdot n \rfloor,$$
  where $dist(\cdot, \cdot)$ is a distance measure, $n = |D|$.

(i.e., an object $o$ is a $DB(r, \pi)$-outlier if the $r$-neighborhood of $o$ has at most $\lfloor \pi \cdot n \rfloor$ other objects)
**DB(r, π)-Outlier Detection Method**

- **Example**: Given $\pi = 0.15$, $n = 15$.
  - We have $\lfloor \pi \cdot n \rfloor = \lfloor 0.15 \cdot 15 \rfloor = \lfloor 2.25 \rfloor = 2$.
  - If the $r$-neighborhood of $o$ has at most 2 neighbors, the object $o$ is a $DB(r, \pi)$-outlier.
  - That is, if the $r$-neighborhood of $o$ has at least 3 neighbors (i.e., $\lfloor \pi \cdot n \rfloor + 1$ or $\lceil \pi \cdot n \rceil$), the object $o$ is not a $DB(r, \pi)$-outlier.
Algorithm: Nested loop algorithm for $DB(r, \pi)$-outlier detection.

Input:
- a set of objects $D = \{o_1, ..., o_n\}$, threshold $r$ ($r > 0$) and $\pi$ ($0 < \pi \leq 1$);

Output: $DB(r, \pi)$ outliers in $D$. 
Nested Loop Algorithm for $DB(r, \pi)$-Outlier

Method:

for $i = 1$ to $n$ do

    $count \leftarrow 0$

    for $j = 1$ to $n$ do

        if $i \neq j$ and $\text{dist}(o_i, o_j) \leq r$ then

            $count \leftarrow count + 1$

            if $count \geq \lceil \pi \cdot n \rceil$ then

                exit \{ $o_i$ cannot be a $DB(r, \pi)$-outlier \}

        endif

    endif

endfor // for $j$
Nested Loop Algorithm for $DB(r, \pi)$-Outlier

```plaintext
print a $DB(r, \pi)$-outlier $o_i$
endfor // for $i$
```

- The time complexity of the nested loop algorithm is $O(n^2)$. 
• Consider a 2-dimensional data set. Data space is partitioned into a 2-dimensional grid, where each cell is a square. The length of each edge of a cell is $r/(2\sqrt{2})$, where $r$ is a distance threshold.

• In $l$-dimensional data space, each cell is a hypercube that has a diagonal of length $r/2$, and the length of each edge of a cell is $r/(2\sqrt{l})$. 
Grid-Based Method for $DB(r, \pi)$-Outlier

\[(\text{diameter} = r, \text{cell length} = \frac{r}{2\sqrt{2}})\]
Grid-Based Method for $DB(r, \pi)$-Outlier

• Consider the cell $C$ shown below. The neighboring cells of $C$ can be divided into two groups.
  - Cells immediately next to $C$ are called the **level-1** cells ($L_1$, labeled by 1), and
  - Cells two or three cells away from $C$ in any direction are called the **level-2** cells ($L_2$, labeled by 2).
Grid-Based Method for $DB(r, \pi)$-Outlier

...
Grid-Based Method for $DB(r, \pi)$-Outlier

- Two levels of cells have the following properties.
  - **Level-1 cell property**: Given any point $x$ in $C$ and any point $y$ in a level-1 cell, then $dist(x, y) \leq r$.
  - **Level-2 cell property**: Given any point $x$ in $C$ and any point $y$ such that $dist(x, y) \geq r$, then $y$ is in a level-2 cell.

- Let $a$ be the number of objects in cell $C$, $b_1$ be the total number of objects in the level-1 cells, and $b_2$ be the total number of objects in the level-2 cells.
Grid-Based Method for $DB(r, \pi)$-Outlier

- We can apply the following rules.

- **Level-1 cell pruning rule:** Based on the level-1 cell property, if $a + b_1 > \lceil \pi \cdot n \rceil$, then every object $o$ in $C$ is not a $DB(r, \pi)$-outlier because all those objects in $C$ and the level-1 cells are in the $r$-neighborhood of $o$, and there are at least $\lceil \pi \cdot n \rceil$ such neighbors.
- **Level-2 cell pruning rule:** Based on the level-2 cell property, if \( a + b_1 + b_2 < \lceil \pi \cdot n \rceil + 1 \), then all objects in \( C \) are \( DB(r, \pi) \)-outliers because each of their \( r \)-neighborhoods has at most \( \lfloor \pi \cdot n \rfloor \) other objects (i.e., less than \( \lceil \pi \cdot n \rceil \) neighbors).
• Using the two pruning rules, the grid-based method organizes objects into groups using a grid—all objects in a cell form a group. For groups satisfying one of the two rules, we can determine that either all objects in a cell are nonoutliers or outliers, and thus do not need to check those objects one by one. Moreover, to apply the two rules, we need only check a limited number of cells close to a target cell instead of the whole data set.
Distance-Based Outlier Detection

• Distance-BOD (e.g., $DB(r, \pi)$-outliers) takes a global view of the data set. Such outliers are called **global outliers** for two reasons:
  - A $DB(r, \pi)$-outlier is far from at least $(1 - \pi) \times 100\%$ of the objects in the data set. That is, an outlier as such is remote from the majority of the data.
  - To detect distance-based outliers, we need two global parameters $r$ (radius) and $\pi$ (percentage), which are applied to every outlier object.
Density-Based Outlier Detection

- Objects may be considered as outliers w.r.t. their local neighborhoods, instead of global data distribution. Such outliers are called **local outliers** detected by a density-based method.
**Example:** Local proximity-based outliers. Consider data points shown below. There are two clusters: \( C_1 \) is dense, and \( C_2 \) is sparse. Object \( o_3 \) can be detected as a distance-based outlier (i.e., **global** outlier) because it is far from the majority of the data set.
Density-Based Outlier Detection

• Consider objects $o_1$ and $o_2$. Distance from $o_1$ and $o_2$ to objects in dense cluster $C_1$ is smaller than average distance between an object in cluster $C_2$ and its nearest neighbors. Thus, $o_1$ and $o_2$ are not distance-based outliers (i.e., not global outliers).

• In fact, if we were to categorize $o_1$ and $o_2$ as $DB(r, \pi)$-outliers, we would have to classify all objects in clusters $C_2$ as $DB(r, \pi)$-outliers.
Density-Based Outlier Detection

- Objects $o_1$ and $o_2$ can be identified as local outliers when they are considered locally w.r.t. cluster $C_1$ because $o_1$ and $o_2$ deviate/differ significantly from the objects in $C_1$. Moreover, $o_1$ and $o_2$ are also far from the objects in $C_2$.
- Distance-BOD methods cannot capture local outliers like $o_1$ and $o_2$. 
• Distance between object $o_4$ and its nearest neighbors is much greater than the distance between $o_1$ and its nearest neighbors. However, because $o_4$ is local to cluster $C_2$ (which is sparse), $o_4$ is not considered a local outlier.
To formulate local outliers, we need to compare the density around an object with the density around its local neighbors.

Basic assumption of density-BOD methods is that density around a nonoutlier object is similar to density around its neighbors, while density around an outlier object is significantly different from density around its neighbors.
Density-Based Outlier Detection

• Density-BOD methods use the relative density of an object against its neighbors to indicate the degree to which an object is an outlier.
• Let us consider how to measure the relative density of an object \( o \), given a set of objects \( D \).
The \textit{k-distance} of \( o \), denoted by \( \text{dist}_k(o) \), is the distance \( \text{dist}(o, p) \) between \( o \) and another object \( p \in D \) such that

- There are at least \( k \) objects \( o' \in D - \{ o \} \) such that \( \text{dist}(o, o') \leq \text{dist}(o, p) \).

- There are at most \( k - 1 \) objects \( o'' \in D - \{ o \} \) such that \( \text{dist}(o, o'') < \text{dist}(o, p) \).

That is, \( \text{dist}_k(o) \) is distance between \( o \) and its \( k \)-th nearest neighbor.
Density-Based Outlier Detection

• The \textit{k-distance neighborhood} of \(o\) contains all objects of which the distance to \(o\) is not greater than \(\text{dist}_k(o)\), denoted by

\[
N_k(o) = \{ o' \mid o' \in D - \{o\}, \text{dist}(o, o') \leq \text{dist}_k(o) \}
\]

• \(N_k(o)\) may contain more than \(k\) objects because multiple objects may each be the same distance away from \(o\).
Density-Based Outlier Detection

- For two objects $o$ and $o'$, the **reachability distance** of $o'$ from $o$ to is defined as

  \[
  \text{reachdist}_k(o', o) = \max \{ \text{dist}_k(o), \text{dist}(o', o) \}
  \]

- $k$ specifies the minimum neighborhood to be examined to determine the local density of an object (i.e., $k = \text{MinPts}$ in DBSCAN algorithm).
Density-Based Outlier Detection

• **Local reachability density** of an object $o$ is defined as

$$lrd_k(o) = \frac{\| N_k(o) \|}{\sum_{o' \in N_k(o)} \text{reachdist}_k(o', o)}.$$

• **Local outlier factor** of an object $o$ is defined as

$$LOF_k(o) = \frac{\sum_{o' \in N_k(o)} \frac{lrd_k(o')}{lrd_k(o)}}{\| N_k(o) \|}.$$
Density-Based Outlier Detection

• The local outlier factor (LOF) of an object $o$ quantifies the degree to which the object $o$ is considered as an outlier.
- If $LOF_k(o) \approx 1$, $o$ is not a local outlier (i.e., inlier).
- If $LOF_k(o)$ is significant greater than 1, $o$ is a local outlier.
Density-Based Outlier Detection

• Two properties of the local outlier factor are

1. An object that is deep inside a cluster (e.g., the points in the center of cluster $C_2$) has the LOF of approximately 1. This property ensures that objects inside dense or sparse clusters are not local outliers.
Density-Based Outlier Detection

2. It can be shown that $LOF(o)$ is bounded as

$$\frac{\text{direct}_{\min}(o)}{\text{indirect}_{\max}(o)} \leq LOF(o) \leq \frac{\text{direct}_{\max}(o)}{\text{indirect}_{\min}(o)}$$
Density-Based Outlier Detection

A property of LOF($o$)
where
- For object \( o \), let

\[
direct_{\text{min}}(o) = \min\{ \text{reachdist}_k(o', o) \mid o' \in N_k(o) \}
\]

be the minimum reachability distance from \( o \) to its \( k \)-nearest neighbors.
- Similarly, we can define

\[
direct_{\text{max}}(o) = \max\{ \text{reachdist}_k(o', o) \mid o' \in N_k(o) \}
\]
- We also consider the neighbors of \( o \)'s \( k \)-nearest neighbors. Let

\[
indirect_{\text{min}}(o) = \min\{reachdist_k(o'', o') \mid o' \in N_k(o) \text{ and } o'' \in N_k(o')\}
\]

and

\[
indirect_{\text{max}}(o) = \max\{reachdist_k(o'', o') \mid o' \in N_k(o) \text{ and } o'' \in N_k(o')\}
\]
Contents

1. Classification-Based Approaches (Supervised, one-class model - SVMs)
2. Clustering-Based Approaches (Unsupervised, CBLOF)
3. Proximity-Based Approaches (Distance-Based: $DB(r, \pi)$, Density-Based: $LOF_k(o)$)
4. Statistical Approaches
   (Data $\not\in [\mu - \sigma, \mu + \sigma]$ is outlier, Grubbs’ Test, Mahalanobis distance)
4. Statistical Approaches

• Statistical methods for outlier detection make assumptions about data normality. They assume that the normal objects in a data set are generated by a stochastic process (a generative model).
• Normal objects occur in regions of high probability for the stochastic model.
• Objects in the regions of low probability are outliers.
4. Statistical Approaches

• The general idea behind statistical methods for outlier detection is to learn a generative model fitting the given data set, and then identify those objects in low-probability regions of the model as outliers.
4. Statistical Approaches

**Example:** Detect outliers using a statistical (Gaussian aka normal) model. The data points except for those in region $R$ fit a Gaussian distribution $g_D$, where for a location $x$ in the data space, $g_D(x)$ gives the probability density at $x$. 

![Diagram showing outliers detection](image)
4. Statistical Approaches

• Gaussian distribution $g_D$ can be used to model the normal data. That is, most of the data points in the data set.
• For each object $y$ in region $R$, we can estimate $g_D(y)$, the probability that this point fits the Gaussian distribution.
• If $g_D(y)$ is very low, $y$ is unlikely generated by the Gaussian model, and thus is an outlier.
Statistical Methods

- Two main types of statistical methods are **parametric** and **nonparametric** (e.g., histogram) methods.
- Parametric methods consider **univariate** data (involving one attribute/variable) and **multivariate** data (involving two or more attributes or variables, Mahalanobis distance, $\chi^2$-statistic can be used).
A **parametric method** assumes that the normal data objects are generated by a parametric distribution with parameter $\Theta = (\mu, \sigma^2)$. The **probability density function** (PDF) of the parametric distribution $f(x, \Theta) = f(x \mid (\mu, \sigma^2))$ gives the probability that object $x$ is generated by the distribution. ($\mu$ is mean, $\sigma^2$ is variance)

- The smaller this value, the more likely $x$ is an outlier.
Normal Distribution $\mathcal{N}(\mu, \sigma^2)$

$$f(x \mid (\mu, \sigma^2)) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$
Normal Distribution $\mathcal{N}(\mu, \sigma^2)$

- Given the values of $\mu$ and $\sigma^2$, we can plot the PDF of normal distribution.
- Given the data points $x_1, x_2, ..., x_n$, we can compute the estimated values of mean $\mu$ and variance $\sigma^2$ using maximum likelihood method.

\[
\mu = \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i
\]

\[
\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2
\]
Nonparametric Methods

- A **nonparametric method** tries to determine the model from the input data.
- Examples of nonparametric methods include **histogram** and **kernel density estimation** (KDE).
- KDE is used to estimate the shape of the function $f(x_1, x_2, \ldots, x_n)$.

$$
\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)
$$

(Gray plot is true distribution)
Example: Univariate outlier detection using maximum likelihood. Suppose a city’s average temperature values in July in the last 10 years are, in value-ascending order, 20.0°C, 24.0°C, 28.1°C, 28.2°C, 28.3°C, 28.4°C, 28.5°C, 29.1°C, 29.2°C, 29.3°C, 29.4°C, 29.5°C, 29.6°C, and 29.7°C. Let’s assume that the average temperature follows a normal distribution $\mathcal{N}(\mu, \sigma^2)$. 
Univariate Outlier Detection

- We compute

\[
\mu = (20.0, 24.0, 28.1, 28.2, 28.3, 28.4, 28.5, 29.1, 29.2, 29.3, 29.4, 29.5, 29.6, 29.7) / 14 = 27.95
\]

\[
\sigma^2 = ((20.0 - 27.95)^2 + (24.0 - 27.95)^2 + (28.1 - 27.95)^2 + (28.2 - 27.95)^2 + (28.3 - 27.95)^2 + (28.4 - 27.95)^2 + (28.5 - 27.95)^2 + (29.1 - 27.95)^2 + (29.2 - 27.95)^2 + (29.3 - 27.95)^2 + (29.4 - 27.95)^2 + (29.5 - 27.95)^2 + (29.6 - 27.95)^2 + (29.7 - 27.95)^2) / 14 \approx 6.75
\]

\[
\sigma = \sqrt{6.75} \approx 2.598 \approx 2.60.
\]
Univariate Outlier Detection

- Most deviating value, $20.0^\circ{\text{C}}$, is $27.95^\circ{\text{C}} - 20.0^\circ{\text{C}} = 7.95^\circ{\text{C}}$ away from estimated mean $\mu = 27.95$.
- We know that the $\mu \pm 3\sigma$ region contains 99.7% data under the assumption of normal distribution.
  
  $3\sigma = 3 \times 2.598 = 7.79$, $\mu - 3\sigma = 27.95 - 7.79 = 20.16$, $\mu + 3\sigma = 27.95 + 7.79 = 35.74$)

- $20.0^\circ{\text{C}}$ is generated by the normal distribution if $7.95 \leq 3\sigma = 7.79$ (i.e., $z$-score $z = |\mu - x|/\sigma = 7.95/\sigma \leq 3$). However, we have $7.95/2.598 \approx 3.059 > 3$. Thus, $20.0^\circ{\text{C}}$ can be identified as an outlier.
Visualization of Univariate Outlier Detection

- **Boxplot** can be used to visualize outliers.
- **Boxplot** method plots univariate input data using a five-number summary.
  - Smallest nonoutlier value Min,
  - Lower quartile $Q_1$ contains lowest 25% of data
  - Median $Q_2$, contains half of data
  - Upper quartile $Q_3$ contains highest 25% of data
  - Largest nonoutlier value Max.
• **Interquantile range** $IQR$ is defined as $Q3 - Q1$.
• Region $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$ contains 99.3% of **nonoutlier** objects.
• **Outlier** objects are outside the region $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$. 
Grubbs’ Test

• *Grubb’s test* (aka *maximum normed residual test*) can be used to detect one outlier at a time in univariate data generated by normal distribution.

• *Grubb’s test* detects one outlier at a time, removes the outlier, and repeat.

• For each object \( x \) in a data set, we define a Grubbs’ test statistic \( G \) (aka *z-score*) as \( G = |x - \mu| / s \), where \( \mu \) is the mean, and \( s \) is the standard deviation of the input data.
Grubbs’ Test

• An object $x$ is an outlier if $G$ is greater than the Grubbs’ critical value $G_c(\alpha, N)$. That is,

$$G > G_c(\alpha, N) = \frac{(N - 1)t_{\alpha/(2N), N-2}}{\sqrt{N(N - 2 + t^2_{\alpha/(2N), N-2})}}$$

where $t_{\alpha/(2N), N-2}$ is the critical value taken by a $t$-distribution with $N - 2$ degrees of freedom and a significance level of $\alpha/(2N)$, and $N$ is the number of objects in the data set.
Grubbs’ Test

• The most commonly used $\alpha = 0.05 \ (5\%)$, $1 - \alpha = 0.95 \ (95\%)$.
• The Grubbs’ test presented above is the two-sided Grubb’s test.
• For the one-sided Grubbs’ test, replace $\alpha/(2N)$ with $\alpha/N$. 
Example
Multivariate Outlier Detection

- **Multivariate** data involves two or more attributes or variables.
- Central idea is to transform multivariate outlier detection task into a univariate outlier detection problem.
- We consider two methods for multivariate outlier detection.
  - Use **Mahalanobis distance**
  - Use $\chi^2$-statistic
Example: Multivariate outlier detection using Mahalanobis distance. For a multivariate data set, let $\bar{o}$ be mean vector of multivariate data set. For an object $o$ in data set, Mahalanobis distance from $o$ to $\bar{o}$ is $MDist(o, \bar{o}) = (o - \bar{o})^T S^{-1} (o - \bar{o})$, where $S$ is a $d \times d$ covariance matrix, $o$ and $\bar{o}$ are $d \times 1$ vectors.

• $MDist(o, \bar{o})$ is a univariate variable, and thus Grubbs’ test can be applied to this measure.
Multivariate Outlier Detection – Mahalanobis

- Multivariate outlier detection task can be performed as follows.
  1. Calculate mean vector $\bar{o}$ of multivariate data set.
  2. For each object $o$, calculate $MDist(o, \bar{o})$.
  3. Detect outliers in transformed univariate data set $\{MDist(o, \bar{o}) \mid o \in D\}$ using Grubbs’ test.
  4. If $MDist(o, \bar{o})$ is an outlier, then $o$ is an outlier.
Example: Multivariate outlier detection using $\chi^2$-statistic. $\chi^2$-statistic (chi-squared) can be used to capture multivariate outliers under the assumption of normal distribution. For an object $o$, $\chi^2$-statistic is defined as

$$\chi^2 = \sum_{i=1}^{n} \frac{(o_i - E_i)^2}{E_i} ,$$
Multivariate Outlier Detection – $\chi^2$-Statistic

where $o_i$ is value of $o$ on $i$th dimension, $E_i$ is mean of $i$-dimension among all objects, and $n$ is dimensionality.

- If $\chi^2$-statistic is large, the object is an outlier.
Mixture of Parametric Distributions

• Read the text book
Example: Outlier detection using a histogram. AllElectronics records purchase amount for every customer transaction. For example, 60% of transaction amounts are between $0.00 and $1000.
• We can use the histogram as a nonparametric statistical model to capture outliers.
  • For example, a transaction in the amount of $7500 is an outlier because only $1 – (60\% + 20\% + 10\% + 6.7\% + 3.1\%) = 0.2\%$ of transactions have an amount higher than $5000$.
  • A transaction amount of $385$ is normal because it falls into the bin (or bucket) holding 60\% of the transactions.
Nonparametric Methods – Histogram

- To determine whether an object $o$ is an outlier, we can check it against generated histogram.
- If the object falls in one of histogram’s bins, the object is normal. Otherwise, it is an outlier.
• It is hard to choose an appropriate bin size.
- If the bin size is set too small, many normal objects may end up in empty or rare bins, and thus be misidentified as outliers. This leads to a high false positive rate and low precision.
- If the bin size is set too high, outlier objects may infiltrate into some frequent bins and thus be disguised as normal. This leads to a high false negative rate and low recall.
Nonparametric Methods – Kernel Function

• To overcome weakness of histogram-based approach, we use kernel density estimation to estimate the probability density distribution of the data.

• Frequently used kernel is a standard Gaussian function with mean 0 and variance 1:

\[
K\left(\frac{x - x_i}{h}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x - x_i)^2}{2h^2}}.
\]
Nonparametric Methods – Kernel Function

- Let $x_1, ..., x_n$ be an independent and identically distributed sample of a random variable $f$.
- The kernel density approximation of the probability density function (PDF) is

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where $K()$ is a kernel and $h$ is the bandwidth serving as a smoothing parameter.
• We can use the estimated density function $\hat{f}$ to detect outliers.

• If $\hat{f}(o)$ is high, then the object is likely normal. Otherwise, $o$ is likely an outlier.
Summary
Exercises
References

1. Jiawei Han, Micheline Kamber, Jian Pei. 2011. *Data Mining Concepts and Techniques*. 3\textsuperscript{rd} Ed. Morgan Kaufmann. ISBN: 9380931913.


• The (arithmetic) **mean**, denoted as $\mu$ or $\bar{x}$, of $n$ values $x_1, x_2, ..., x_n$ is defined as

$$
\mu = \bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i
$$

• The **harmonic mean**, denoted as $H$, of $n$ positive real numbers $x_1, x_2, ..., x_n$ is defined as

$$
H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... + \frac{1}{x_n}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}
$$
• The **variance**, denoted as $\sigma^2$ or $s^2$, of a random variable $x = (x_1, x_2, ..., x_n)$ is defined as

$$\sigma^2 = s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu)^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 = Var(x)$$

(unbiased sample variance)

https://en.wikipedia.org/wiki/Variance

• **Example:**

$x = [4.00 \ 4.20 \ 3.90 \ 4.30 \ 4.10]$; // Length

$V = \text{var}(x)$

$V = 0.0250$
Extra Slides – Standard Deviation

- The (population) **standard deviation**, denoted as \( \sigma \) or \( s \), of \( n \) values \( x_1, x_2, \ldots, x_n \) is defined as

\[
\sigma = s = \sqrt{\sigma^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu)^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}
\]


```plaintext
>> x = [10 7 9];
>> SD = std(x)
>> SD = 1.5275
```
• The probability density function (PDF) of the normal distribution \( \mathcal{N}(\mu, \sigma^2) \) is

\[
f(x \mid (\mu, \sigma^2)) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}
\]

where \( \mu \) is mean, \( \sigma^2 \) is variance, \( \sigma \) is standard deviation.
Normal distribution $\mathcal{N}(\mu, \sigma^2)$, bell curve
(This fact is known as the 68-95-99.7 (empirical) rule, or the 3-sigma rule.)
Extra Slides – Normal Distribution
Kernel density estimation (KDE) is a non-parametric way to estimate the probability density function (PDF) of a random variable.

Let \((x_1, x_2, \ldots, x_n)\) be a univariate independent and identically distributed sample drawn from some distribution with an unknown density \(f\). We are interested in estimating the shape of this function \(f\).
The kernel density estimator (KDE) of $f$ is

$$
\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)
$$

where $K$ is the kernel (i.e., a non-negative function) and $h > 0$ (e.g., $h = 0.337$) is a smoothing parameter called the bandwidth. A kernel with subscript $h$ is called the scaled kernel and defined as $K_h(x) = 1/h K(x/h)$. 
Grey plot is true density of standard normal distribution
**Extra Slides – Grubbs’ Critical Values $G_c$ Table**

<table>
<thead>
<tr>
<th>$n$</th>
<th>$G_c$</th>
<th>$n$</th>
<th>$G_c$</th>
<th>$n$</th>
<th>$G_c$</th>
<th>$n$</th>
<th>$G_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1.1543</td>
<td>12</td>
<td>2.4116</td>
<td>3</td>
<td>1.1531</td>
<td>12</td>
<td>2.2850</td>
</tr>
<tr>
<td>4</td>
<td>1.4812</td>
<td>13</td>
<td>2.4620</td>
<td>4</td>
<td>1.4625</td>
<td>13</td>
<td>2.3305</td>
</tr>
<tr>
<td>5</td>
<td>1.7150</td>
<td>14</td>
<td>2.5073</td>
<td>5</td>
<td>1.6714</td>
<td>14</td>
<td>2.3717</td>
</tr>
<tr>
<td>6</td>
<td>1.8871</td>
<td>15</td>
<td>2.5483</td>
<td>6</td>
<td>1.8221</td>
<td>15</td>
<td>2.4090</td>
</tr>
<tr>
<td>7</td>
<td>2.0200</td>
<td>16</td>
<td>2.5857</td>
<td>7</td>
<td>1.9381</td>
<td>16</td>
<td>2.4433</td>
</tr>
<tr>
<td>8</td>
<td>2.1266</td>
<td>17</td>
<td>2.6200</td>
<td>8</td>
<td>2.0317</td>
<td>17</td>
<td>2.4748</td>
</tr>
<tr>
<td>9</td>
<td>2.2150</td>
<td>18</td>
<td>2.6516</td>
<td>9</td>
<td>2.1096</td>
<td>18</td>
<td>2.5040</td>
</tr>
<tr>
<td>10</td>
<td>2.2900</td>
<td>19</td>
<td>2.6809</td>
<td>10</td>
<td>2.1761</td>
<td>19</td>
<td>2.5312</td>
</tr>
<tr>
<td>11</td>
<td>2.3547</td>
<td>20</td>
<td>2.7082</td>
<td>11</td>
<td>2.2339</td>
<td>20</td>
<td>2.5566</td>
</tr>
</tbody>
</table>

($\alpha = 0.05$)
Extra Slides