Estimation, Detection and Learning II

Outliers Detection

Catarina Oliveira

DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

UPT UNIVERSIDADE PORTUCALENSE

CONTENT

1. Statistical methods
2. proximity methods
3. *clustering*

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

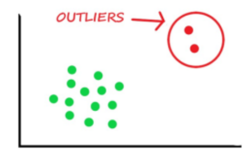
Outliers

Outlier : observation that deviates significantly from the rest of the observations (other than noise)

Applications: fraud detection, medicine, security, industry, image processing, video/sensor network surveillance, intrusion detection

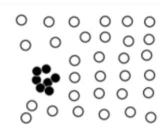
types of *outliers*

Global :
- Observation that deviates significantly from the other
- Simplest type of *outlier*
- Most methods aim to detect this type



Contextual / Conditional :
- Observation that deviates significantly from the others in a given context (ex : the temperature today is 30ºC – in December it is an *outlier* ; in July it is not)
- The context has to be specified along with the problem definition
- Attributes separated into two types :
  - Contextual : define the context (eg date, location)
  - Behavioral : define the characteristics of the object, being used to assess whether or not it is an *outlier* in the context to which it belongs (eg temperature, humidity)

Collective :
- Set of objects that deviate significantly from the rest
- Each object alone is not an *outlier*



Outlier

Modeling normal objects and *outliers* effectively

The boundary between data normality and abnormality (*outliers*) is usually not well defined

data dependent

Ex: in medicine, a small variation can be significant; in marketing it would take a large variation to be meaningful

deal with the noise

It is necessary to remove the noise before detecting *outliers* to avoid the *outlier* being “masked” by the noise

comprehensibility

Justify the detected *outlier*

Outlier

Supervised

Experts identify data as being normals/ *outliers* and later it can be seen as a classification problem

Challenges:

- Unbalanced classes (normal data are much larger than *outliers*)
- Finding as many *outliers* as possible is more important than not misclassifying normals as *outliers.*

Unsupervised

We don't know which objects are normal/ *outliers*

Objects are assumed to be " *clustered* " and *outliers* are further away

semi-supervised

Similar to *supervised* , but with only a subset of the data identified as normals/ *outliers*

Outlier

Statistics (*model-based*)

They assume: data generated by a statistical model (stochastic); data that do not follow the model are outliers

Proximity

Assume: an object is an *outlier* if its nearest neighbors are far from the *feature space* (ie : the proximity of the object to its neighbors deviates from the proximity of most objects to its neighbors)

clustering

Assume: normal objects belong to dense and large *clusters and outliers* belong to small or sparse *clusters or do not belong to any cluster*

Statistical methods

Statistical methods

They assume that the normal objects in a dataset are generated by a stochastic process:

- Normal objects occur in high probability regions for the stochastic model
- Objects in low probability regions are *outliers*

Two categories:

- **Parametric** : assume that normal objects are generated by a parameterized parametric distribution Θ. The *probability density function* of the parametric distribution $f(x, \Theta)$determines the probability of $x$being generated by that distribution. The smaller this value, x the more likely it is to be an *outlier.*
- **Non-parametric** : assume no a priori statistical model, but try to determine the model from the input data

Parametric statistical methods

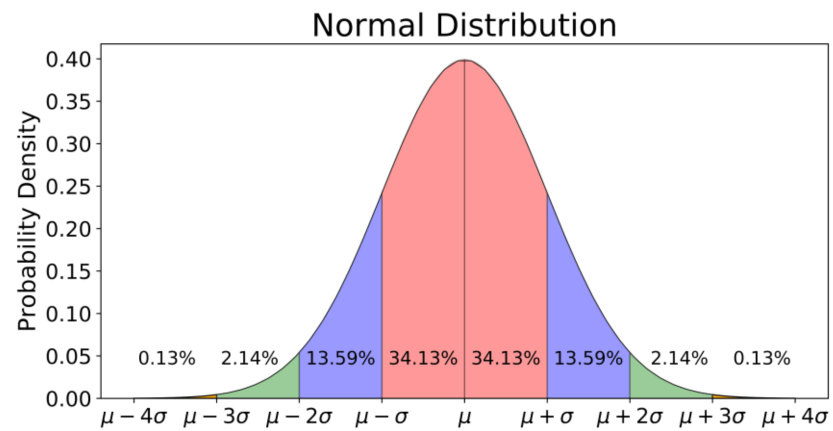
Univariate data : assume normal distribution – use *maximum likelihood*

Example:

Considering a sample of *n* ordered values (eg : 24.0; 28.9; 28.9; 29.0; 29.1; 29.1; 29.2; 29.2; 29.3; 29.4)

Assuming that the values follow the normal distribution with mean μ and standard deviation σ

maximum is obtained *likelihood estimates* :



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

In the example = 28.61

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

In the example ≈ 2.29 $\hat{\sigma} \approx \sqrt{2{,}29} \approx 1{,}51$

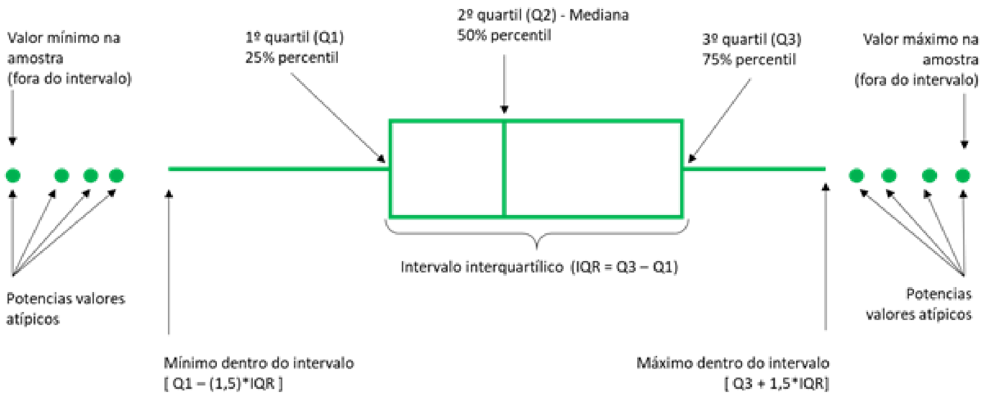
In the normal distribution, $\mu \pm 3\sigma$it contains approximately 97.7% of the data

A value outside this range is likely to be an *outlier* . That is, the values v : $\frac{\mu - v}{\sigma} > 3$, because the probability of following the same distribution is < 0.15%

In the example, 24.0 is an *outlier* , because$\frac{28{,}61 - 24{,}0}{1{,}51} \approx 3{,}05 > 3$

UPT DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Univariate data : assume normal distribution – use *boxplot*



Outliers are the values v :

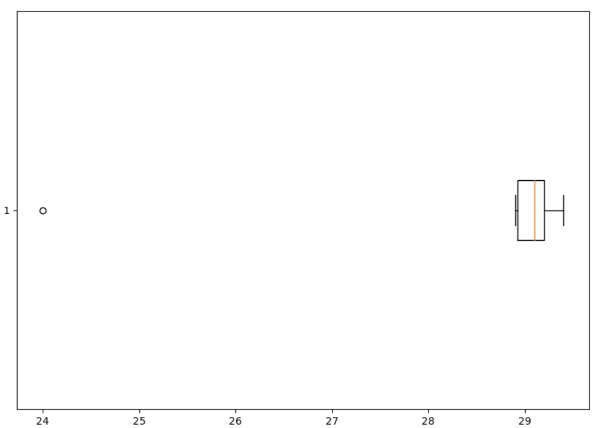
$v < Q1 - 1{,}5 \times IQR$

or

$v > Q3 + 1{,}5 \times IQR$

Example:

Considering a sample of *n* ordered values (eg : 24.0; 28.9; 28.9; 29.0; 29.1; 29.1; 29.2; 29.2; 29.3; 29.4)




```
import matplotlib.pyplot as plt
fig = plt.figure ( figsize =(10, 7))
plt.boxplot (x=[24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29.4], vert
=False)
plt.show ()
```

24.0 is an outlier

Univariate data : assume normal distribution – use *Grubb test*

For each object x in a set of N values with mean $\bar{x}$ and standard deviation s, we define its z-score:

$$z = \frac{|x - \bar{x}|}{s}$$

The object x is an *outlier* if:

$z \geq \frac{N-1}{\sqrt{N}} \sqrt{\frac{{t^2}_{\alpha/(2N),N-2}}{N-2+{t^2}_{\alpha/(2N),N-2}}}$, on what:

${t^2}_{\alpha/(2N),N-2}$ is the value following a distribution t with a significance level $\alpha/(2N)$ of $N-2$ degrees of freedom

Multivariate data : assume normal distribution, make *univariate* – *use Mahalanobis*

Let be $\bar{o}$the average vector of a *dataset* $D$and $S$the covariance matrix.

For each object $o$in the *dataset* , *the Mahalanobis* distance from oa $\bar{o}$is:

$$MDist(o,\bar{o}) = (o-\bar{o})^{T}S^{-1}(o-\bar{o})$$

$MDist(o,\bar{o})$is a *univariate variable, and Grubb* 's test can be applied .

outlier detection into *multivariate* data as follows:

1. Calculate the average vector of the *dataset*
2. For each object $o$calculate$MDist(o,\bar{o})$
3. Detect *outliers* in *the dataset* transformed $\{MDist(o,\bar{o}), o \in< D\}$
4. If it is determined to $MDist(o,\bar{o})$be an *outlier* then $o$it is also an *outlier*

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Multivariate data : assume normal distribution, make *univariate* – use Chi square

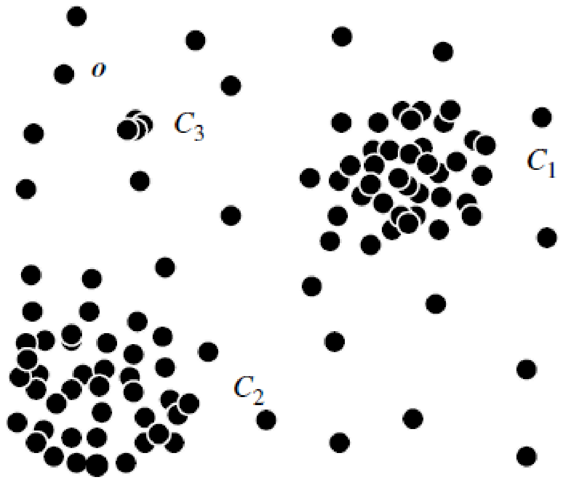
For each object o in a *dataset* with n observations, the chi square is:

$$\chi^2 = \sum_{i=1}^{n} \frac{(o_i - E_i)^2}{E_i}$$

, is o_i the value of o nth i dimension ; E_i is the mean of the i- th dimension

If the value of χ^2 is high, the object o is an *outlier* .

Multivariate data : assume multiple normal distributions



Considering the data in the figure, there are two clusters.
Assuming that the data are generated by a normal distribution would estimate the mean in the middle of the two clusters, and objects between the clusters would not be detected as *outliers* .

We can assume that the normal objects are generated by several normal distributions.
In this case, with 2 distributions, we assume normal distributions:
$\Theta_1(\mu_1, \sigma_1)$It is$\Theta_2(\mu_2, \sigma_2)$

For each object *o* in the *dataset* , the probability that *o* is generated by composing the two distributions is:

$\Pr(o|\Theta_1, \Theta_2) = f_{\Theta_1}(o) + f_{\Theta_2}(o)$, $f_{\Theta_1}$and $f_{\Theta_3}$are the *probabilities density functions* of $\Theta_1$and Θ_2, respectively.

We can use the *Expectation algorithm Maximization* (EM) [(1)] to get the parameters μ_1, σ_1, μ_2 *e* σ_2.

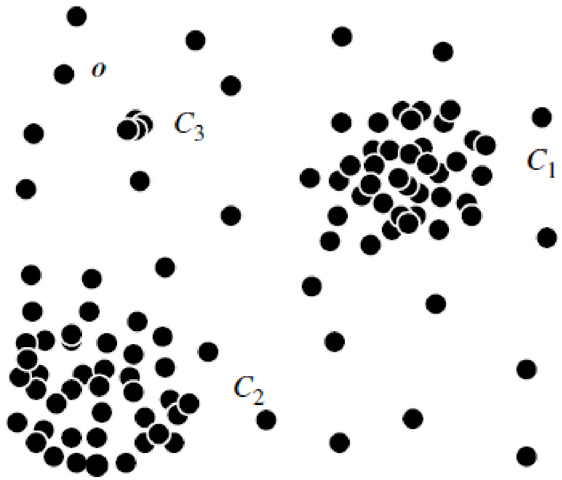
o object is an *outlier* if it does not belong to any cluster

[(1)] https://scikit-learn.org/stable/modules/mixture.html

DEPARTAMENTO **CIÊNCIA E TECNOLOGIA**

Multivariate data : use multiple clusters

Cluster C3 must be detected as *an outlier*



We can assume that normal objects are generated by a normal distribution, or a composite of normal distributions, and that *outliers* are generated by another distribution.

For example, we can assume that this distribution has a greater variance if the *outliers* are distributed over a larger area.

In practice, we define $\sigma_{outlier} = k\sigma$, where *k* is a user-defined parameter and $\sigma$is the standard deviation of the normal distribution that generates the data.

We can also use the EM algorithm

DEPARTAMENTO **CIÊNCIA**
E TECNOLOGIA

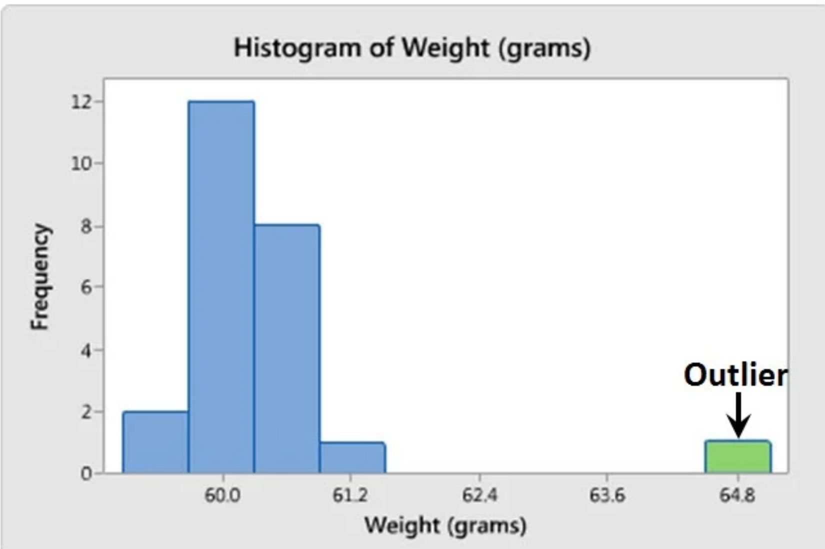
Non-parametric statistical methods

histogram

Procedure:

1. Build the histogram from the data
2. Determine the *outliers* : objects that belong to the least populated " *bins* " or the most "far away"

Example:



proximity methods

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

proximity methods

Given a set of objects in a *feature space* , a distance measure can be used to quantify the similarity between objects. Objects further away can be considered *outliers* .

outlier 's proximity to its nearest neighbors significantly deviates from the object's proximity to most other objects in the set

Two types of methods:

- ***Distance-based*** : queries the neighborhood of an object, defined by a given radius. An object is considered *an outlier* if its neighborhood does not have enough points
    - *Outliers* (taking into account the entire dataset)
- ***Density-based*** : investigates the density of an object and its neighbors. An object is an *outlier* if its density is much lower than that of its neighbors.
    - Allows local *outliers* (taking into account local neighborhoods)

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

proximity methods

Distance

DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Detection of *outliers* by proximity - distance

in a *dataset D of objects, a* distance *threshold , r,* is defined for the neighborhood of an object

For each object, *the* , check the number of objects in its *r* -neighborhood

If most objects in *D* are far from *o* (not in its *r* -neighborhood), then *o* is an *outlier*

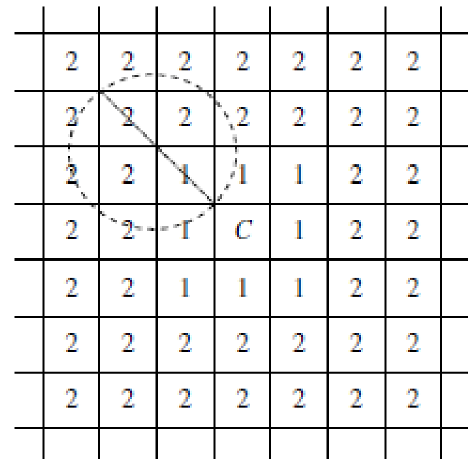
be π $(0 < \pi < 1)$a *threshold* (fraction). An *o -object* is an $DB(r, \pi)$-outlier *if* :

$$\frac{\|\{o' | dist(o, o') \leq r\}\|}{\|D\|} \leq \pi$$

(within radius neighborhood $r$there are less than πobjects)

Outlier detection by proximity – distance – grid (CELL method)

Feature space is partitioned into a multidimensional grid, where each cell is a "hypercube" with a diagonal of size $\frac{r}{2}$, where *r* is the distance *threshold* . If the dataset has *l* dimensions, the edge size of each cell will be $\frac{r}{2\sqrt{l}}$



Considering a 2D dataset , the edge length of each cell is $\frac{r}{2\sqrt{2}}$

Cell C has the elements; b_1 It is b_2 are the total number of elements in the cells marked with 1 and 2, respectively

The neighboring cells of C can be divided into 2 groups of different levels:

- **Level 1** - adjacent to C
    - Given any point $x \epsilon C$ and any possible point y in a level 1 cell, then $dist(x, y) \leq r$
    - If $a + b_1 > \lceil \pi n \rceil$, all o objects *of* C are not $DB(r, \pi)$- *outliers* , because all the objects of C and of the level 1 cells are in the *r* -neighborhood of *o* and there are at least $\lceil \pi n \rceil$ neighbors with these characteristics
- **Level 2** - at a distance of 1 or 2 cells from C
    - Given any point $x \epsilon C$ and any possible point y such that $dist(x, y) \geq r$, then y it is in a level 2 cell
    - If $a + b_1 + b_2 < \lceil \pi n \rceil + 1$, all C objects are $DB(r, \pi)$-outliers , because each of its *r* - neighborhoods has less than $\lceil \pi n \rceil$ objects

proximity methods

Density

Outlier detection by proximity – density – local proximity

Assume: relative density (surrounding) of a normal object is significantly different from the relative density of its neighbors

Given an object *o* and a set of objects *D* , the distance- *k* , $dist_k(o)$, is the distance $\mathrm{dist}(o,p)$between objects *o* and *p* such that:

- There are at least *k* objects $o' \in D - \{o\}$such that$\mathrm{dist}(o,o') \leq dist(o,p)$
- There are at most *k-1* objects $o'' \in D - \{o\}$such that$\mathrm{dist}(o,o'') < dist(o,p)$

That is, $dist_k(o)$it is the distance between *o* and its *k* nearest neighbors

The *k- distance - neighborhood* of *o* contains all objects whose distance to *o* is not greater than $dist_k(o)$.

The local density of *o* is the average of the distances from o to objects in the *k- distance - neighborhood* of *o*

clustering

Outlier detection with *clustering*

After running the *clustering* , let's check what the *outliers are* .

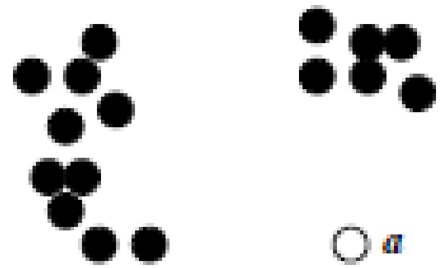
Outliers :

- Objects that do not belong to any cluster
- Objects that are far from the nearest cluster
- Objects that are part of a small or sparse cluster

Outlier detection with *clustering* : objects that do not belong to any *cluster*

Using a *clustering algorithm density-based* , (ex : DBSCAN) we were able to determine that:

- Black dots belong to clusters
- The white dot *a* does not belong to any cluster
    - it is an outlier



Outlier detection with *clustering* : objects far from the nearest

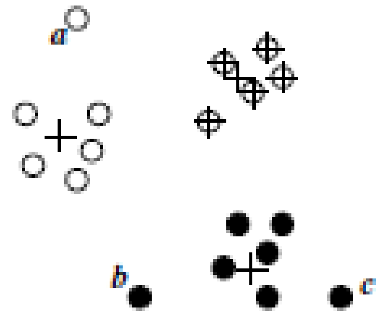
Using, for example, *k- means* we can partition the data into 3 clusters

different symbols

The center of each cluster is marked with +

can assign a score to each object according to the distance between the object and the nearest centroid and compare this distance with the other elements of the cluster

If there is a very large difference, the object is an *outlier.*

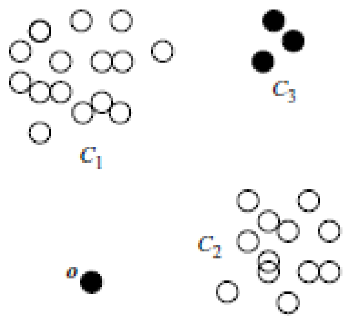


UPT DCT DEPARTAMENTO CIÊNCIA E TECNOLOGIA

Outlier detection with *clustering* : objects in small

Using Cluster- based Local Outlier Factor (CBLOF) we were able to identify *the* and the objects in the C3 cluster as *outliers*

Considers the similarity between the object and the points of the *clusters*



UNIVERSIDADE PORTUCALENSE

Do conhecimento à prática.