COMP 6838 Data Mining

CLASS 5:
Data preprocessing: Outlier Detection
(novelty detection)

Dr. Edgar Acuna
Department of Mathematics

University of Puerto Rico- Mayaguez

math.uprm.edu/~edgar

An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism (Hawkins, 1980).

A comprehensive treatment of outliers in the field of statistics appears in Barnet and Lewis (1994). They provide a large list of outlier detection methods.These methods have two main drawbacks:

First, almost all of them are for univariate data.

Second, all of them are distribution-based. Real-world data are commonly multivariate with unknown distribution.

People in the data mining community got interested in outliers after Knorr and Ng (1998) proposed a non-parametric approach to outlier detection based on the distance of a instance to its nearest neighbors.

Outlier detection a.k.a novelty detection has many applications among them: Fraud detection and network intrusion.

Frequently, outliers are removed to improve accuracy of the estimators. However, this practice is not recommendable because sometimes outliers can have very useful information.

Univariate outliers

Let $\overline{x}$ be the mean and let *s* be standard deviation of the data distribution. One observation is declared as an outlier if lies outside of the interval

$$(\overline{x} - ks, \overline{x} + ks) \quad (1)$$

where the value of k is usually taken as 2 or 3. The justification of these values relies on the fact that assuming normal distribution one expects to have a 95.45% (99.75%, respectively) percent of the data on the interval centered in the mean with a semi-length equal to two (three, respectively) standard deviation.

From equation (1), the observation x is considered an outlier if

$$\frac{|x - \overline{x}|}{s} > k \quad (2)$$

- rownames(bupa[abs(zbupa1)>2,])
- [1] "38" "69" "97" "122" "155" "212" "224" "237" "244" "323" "333"

The problem with the above criteria is that it assumes normal distribution of the data something that frequently does not occur. Furthermore, the mean and standard deviation are highly sensitive to outliers.

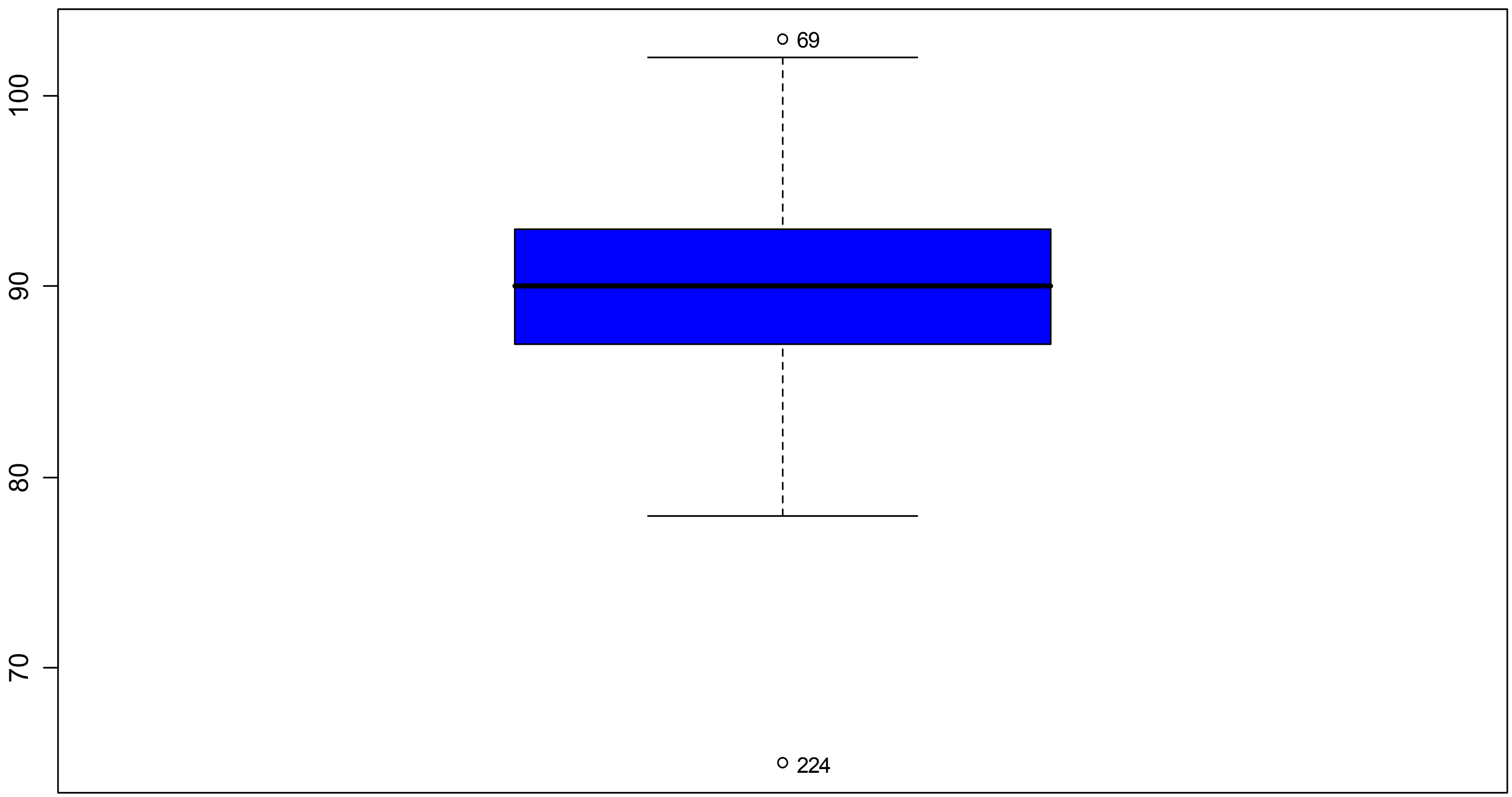
The ***Boxplot*** (Tukey, 1977) is a graphical display for exploratory data analysis, where the outliers appear tagged. Two types of outliers are distinguished: ***mild outliers*** and ***extreme outliers***.

An observation x is declared an ***extreme outlier*** if it lies outside of the interval (Q_1-3×IQR, Q_3+3×IQR),where IQR=Q_3-Q_1 is called the ***Interquartile Range***. An observation x is declared *a* ***mild outlier*** if it lies outside of the interval (Q_1-1.5×IQR, Q_3+1.5×IQR).

The numbers 1.5 and 3 are chosen by comparison with a normal distribution.

Drawing a Boxplot

```
outliers=boxplot(bupa$V1,plot=F)$out
nout=as.character(outliers)
boxplot(bupa$V1,col="blue")
for(i in 1:length(outliers))
{
text(outliers[i],as.character(which(bupa$V1==outlier
s[i])),cex=.8,pos=4)
}
```



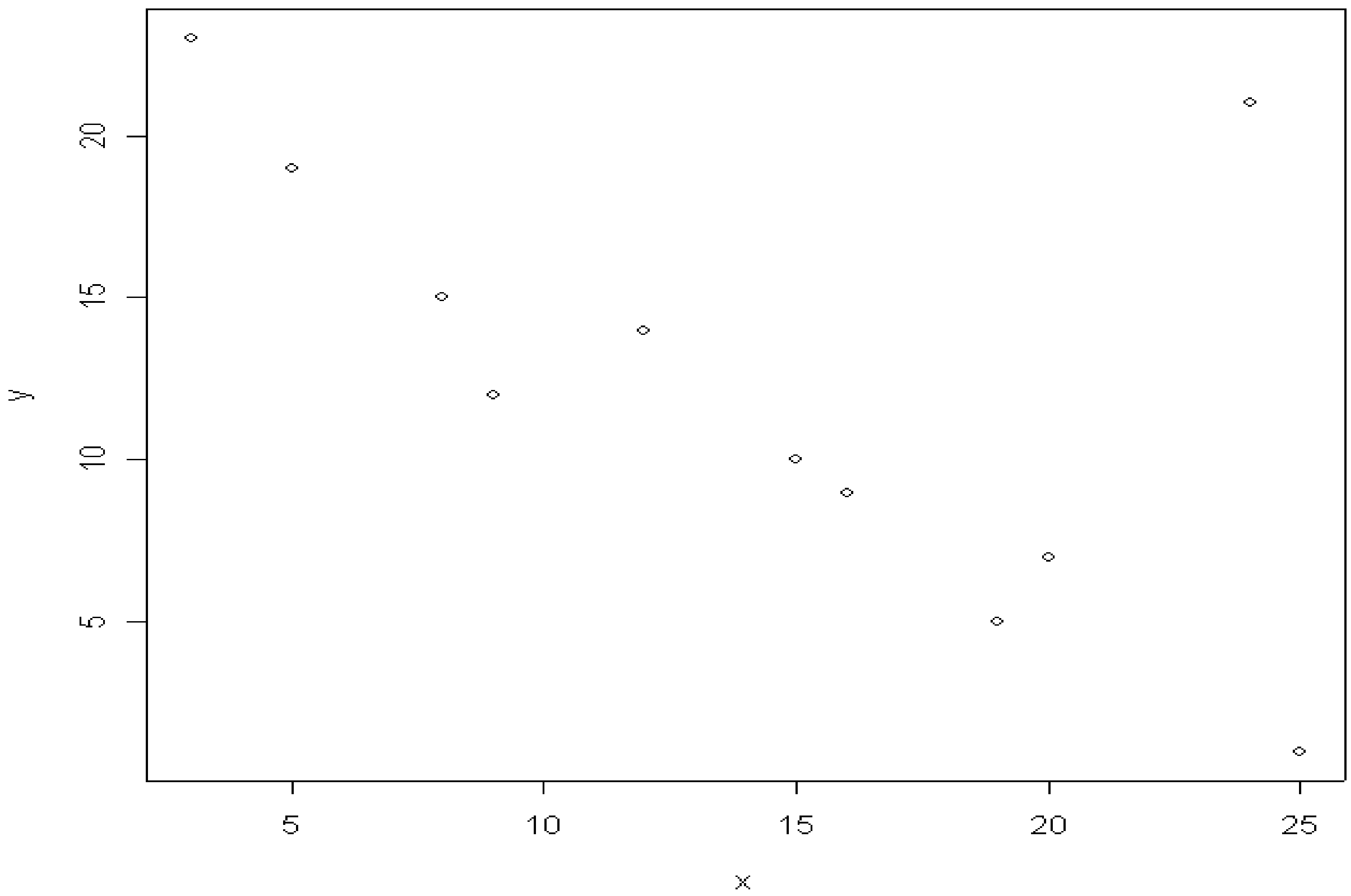
100
90
80
70
69
224

Multivariate Outliers

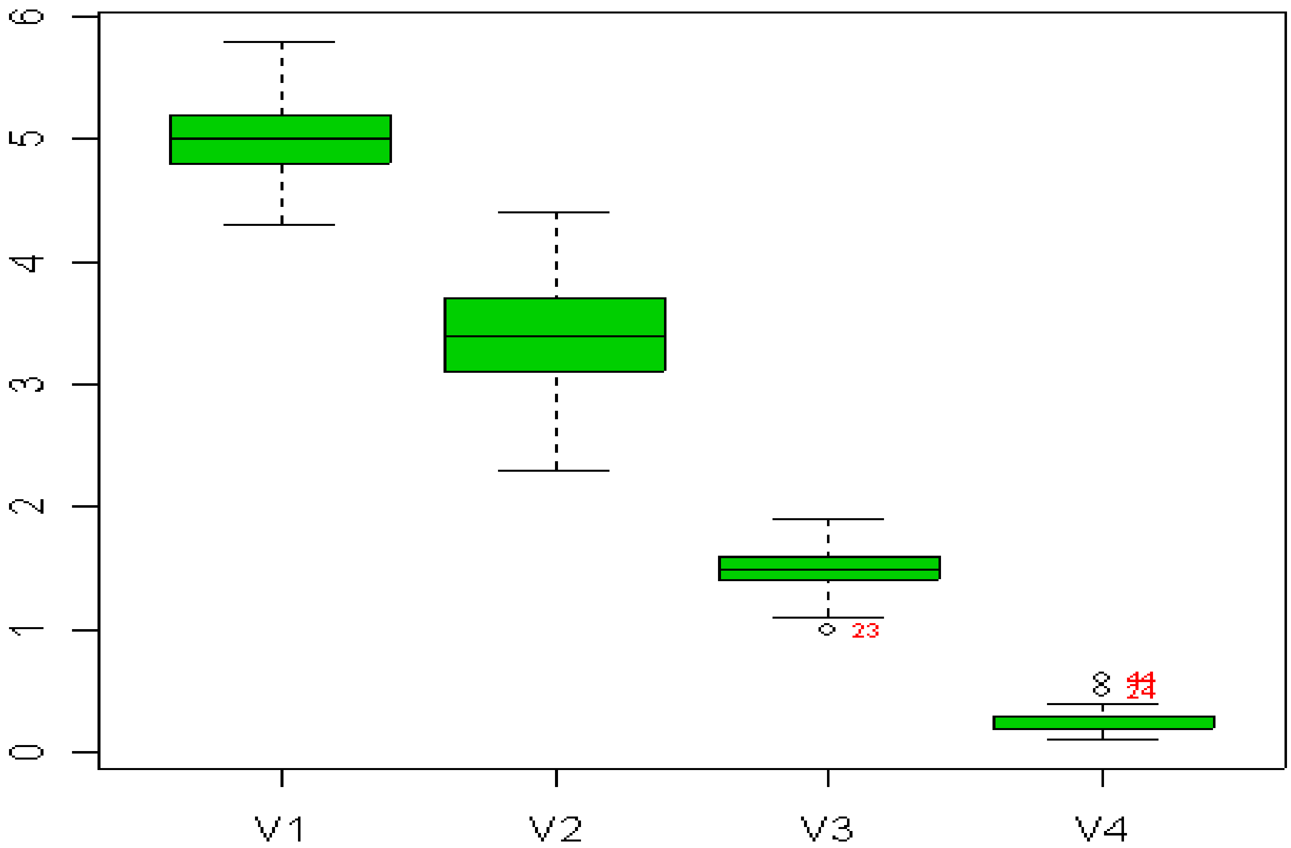
Let us consider a dataset D with p features and n instances. In a supervised classification context, we must also know the classes to which each of the instances belongs.

The objective is to detect all the instances that seem to be unusual, these will be the multivariate outliers.

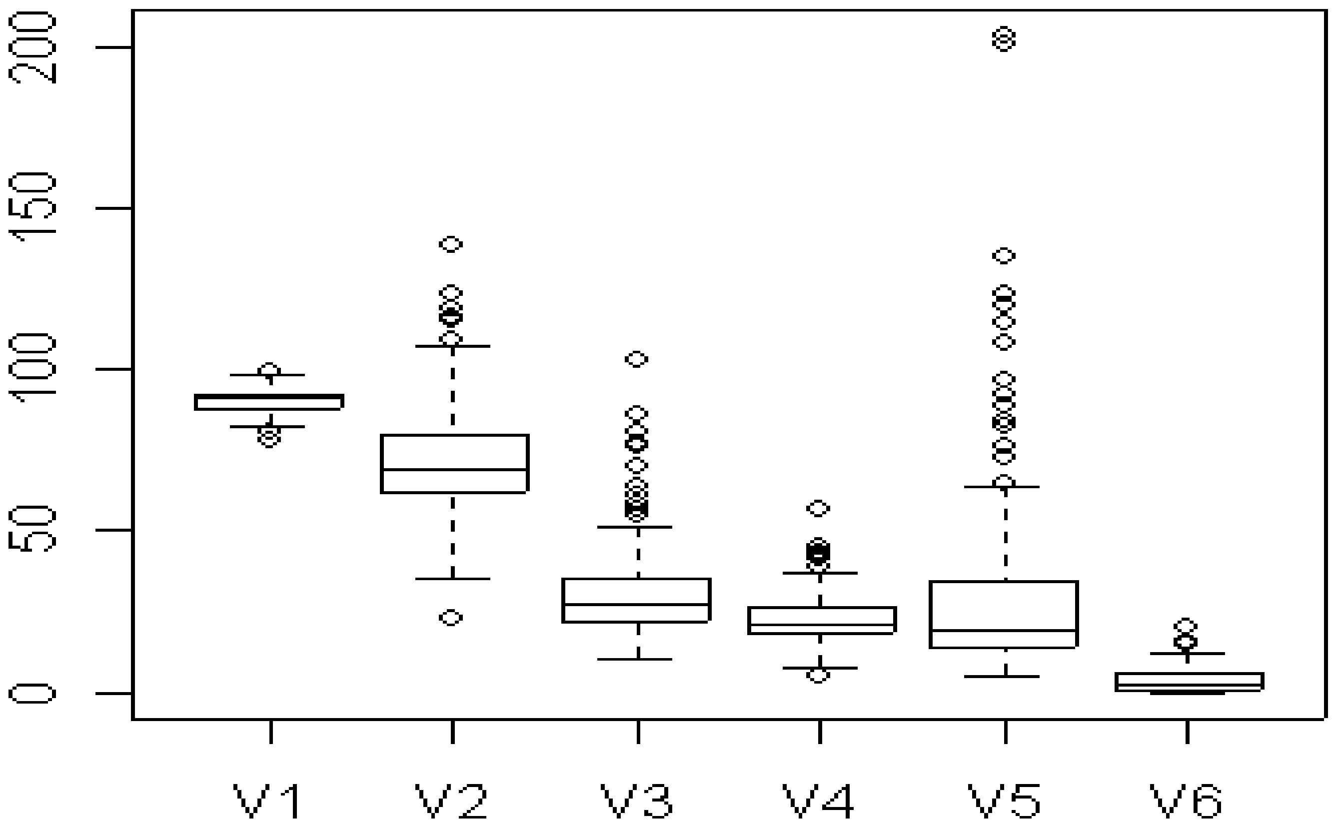
One might think that multivariate outliers can be detected based on the univariate outliers in each feature, but this is not true. On the other hand, an instance can have values that are outliers in several features but the whole instance might not be a multivariate outlier.



A bi-dimensional outlier that is not an outlier in either of its projections.



Detecting outliers for the four features in class 1 of the *Iris* data set



Outliers in the 6 features of the first class for the Bupa dataset

Methods to detect multivariate outliers

- Robust statistical-based outlier detection,
- Outlier detection by clustering,
- Distance-based outlier detection, and
- Density-based local outlier detection.

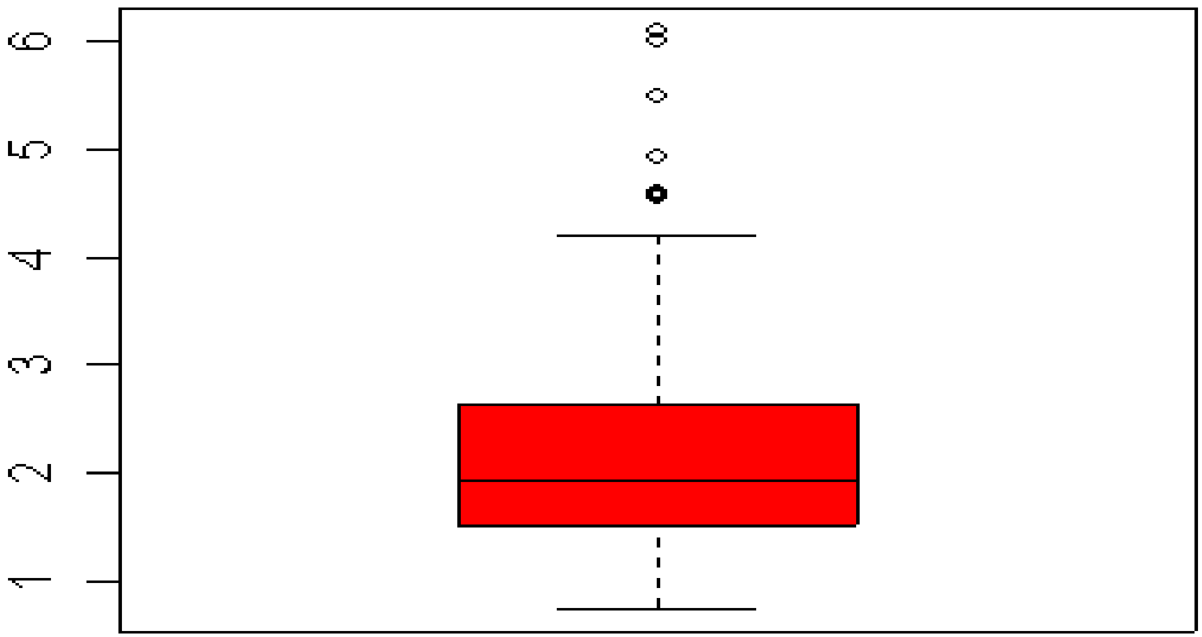
Robust Statistical based outlier detection

Let **x** be an observation of a multivariate data set consisting of n observations and p features. Let $\bar{\mathbf{x}}$ be the centroid of the dataset, which is a p-dimensional vector with the means of each feature as components. Let **X** be the matrix of the original dataset with columns centered by their means. Then, the p×p matrix **S=**1/(n-1) **X'X** represents the covariance matrix of the p features. The multivariate version of equation (2) is

$$D^2(\mathbf{x},\bar{\mathbf{x}}) = (\mathbf{x}-\bar{\mathbf{x}})'\mathbf{S}^{-1}(\mathbf{x}-\bar{\mathbf{x}}) > k$$

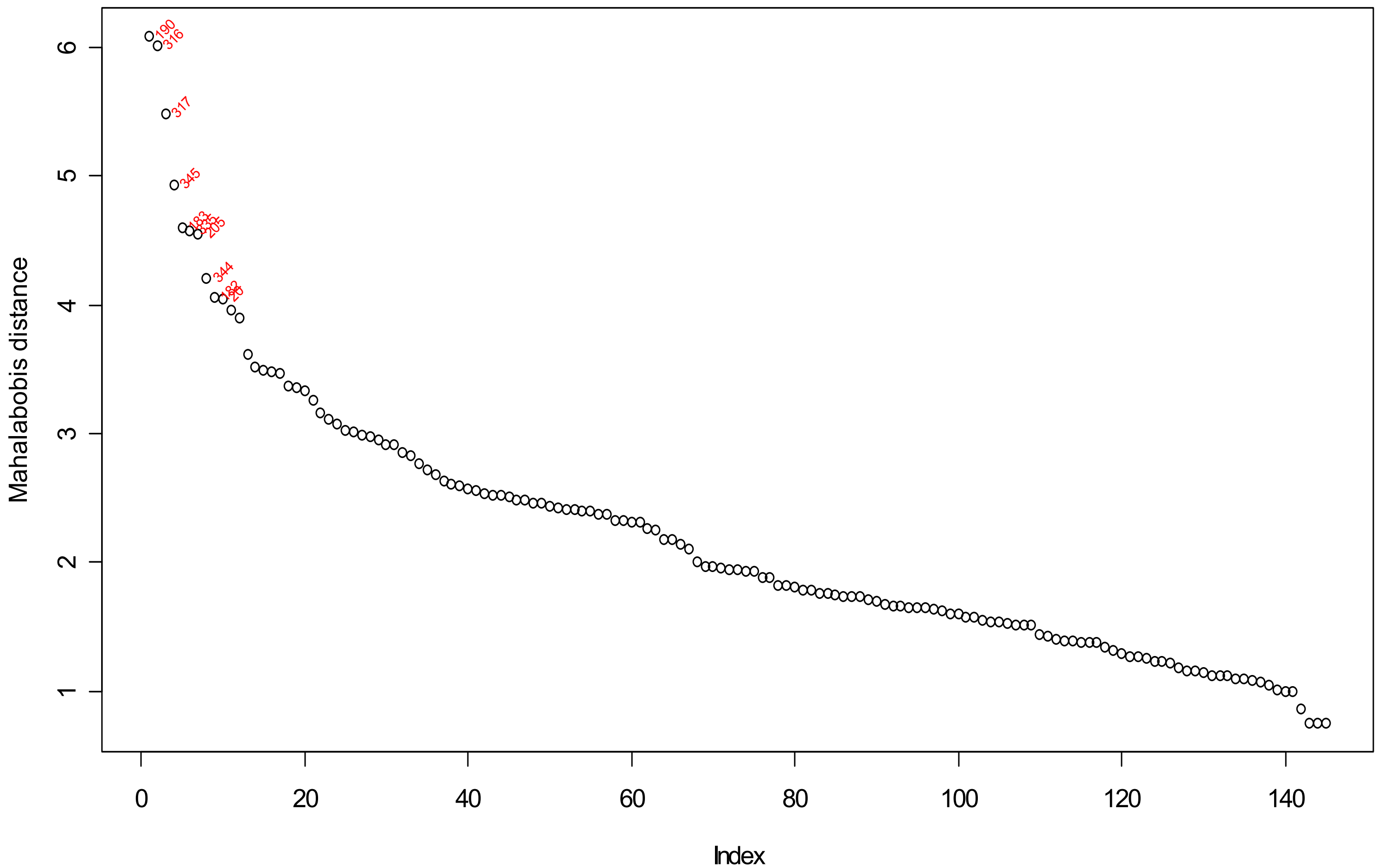
where D^2 is called the Mahalanobis square distance from **x** to the centroid of the dataset. An observation with a large Mahalanobis distance can be considered as an outlier.

```
a=mahaout(bupa,1,T)
Ouliers given by the boxplot of the  Mahalanobis distance
     190      316      317      345      183      335      205 
6.086927 6.012780 5.485214 4.923153 4.593790 4.570818 4.545537 

> boxplot(a)$out
     190      316      317      345      183      335      205 
6.086927 6.012780 5.485214 4.923153 4.593790 4.570818 
4.545537 
> boxplot(a,col="red")
>
```



Boxplot for the Mahalanobis distance for all instances in class 1 of Bupa

Outliers in Bupa (class1) according to Mahalanobis distance



Two effects of multivariate outliers

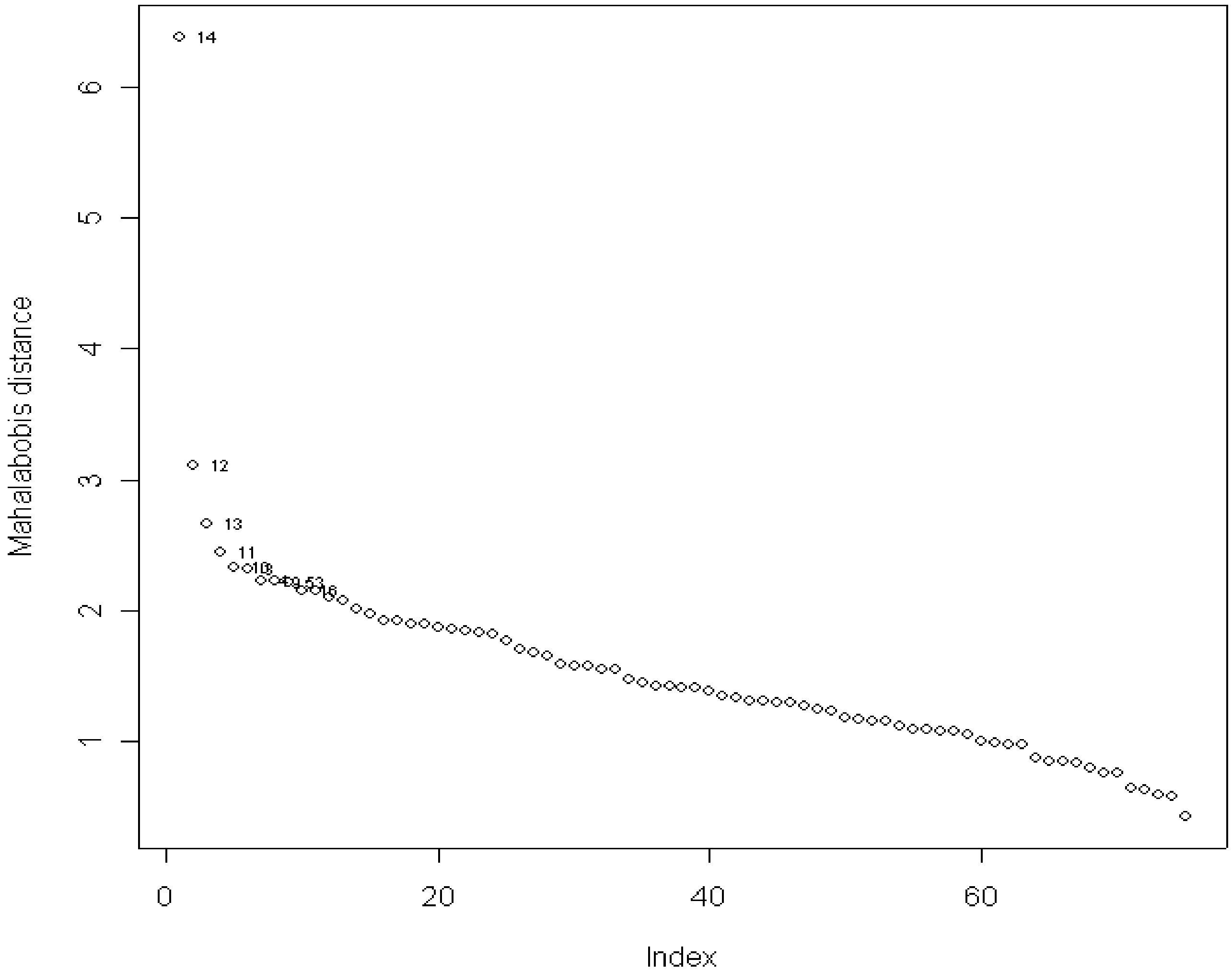
Masking effect. It is said that an outlier masks a second one that is close by if the latter can be considered an outlier by itself, but not if it is considered along with the first one.
Equivalently after the deletion of one outlier, the other instance may emerge as an outlier.

Swamping effect. It is said that an outlier swamps another instance if the latter can be considered outlier only under the presence of the first one. In other words after the deletion of one outlier, the other outlier may become a “good” instance.

To deal with these effects a robust estimator of the Mahalanobis distance is recommended.

The Hawkins-Bradu-Kass dataset

- It has 4 attributes(3 predictors and one response) and 75 instances. The first 14 instances are contaminated to turn them in outliers.



The Masking effect of multivariate outliers in the Hawkins data set (only one outlier out of 14 are detected)

Robust estimator of multivariate location and covariance matrices

- ***The Minimum Volume Elipsoid*** (MVE) estimator, Rousseeuw(1983).
- The ***Minimun Covariance Determinant*** (MCD) estimator, Rousseeuw (1983).
- ***The Donoho-Stahel estimator (1981).***

The ***Minimum Volume Elipsoid (MVE)*** *estimator* is the center and the covariance of a subsample size h ($h \le n$) that minimizes the volume of the covariance matrix associated to the subsample. Formally,

$$\text{MVE}=(\ \bar{\mathbf{x}}_J^*, S_J^*\ )$$

where

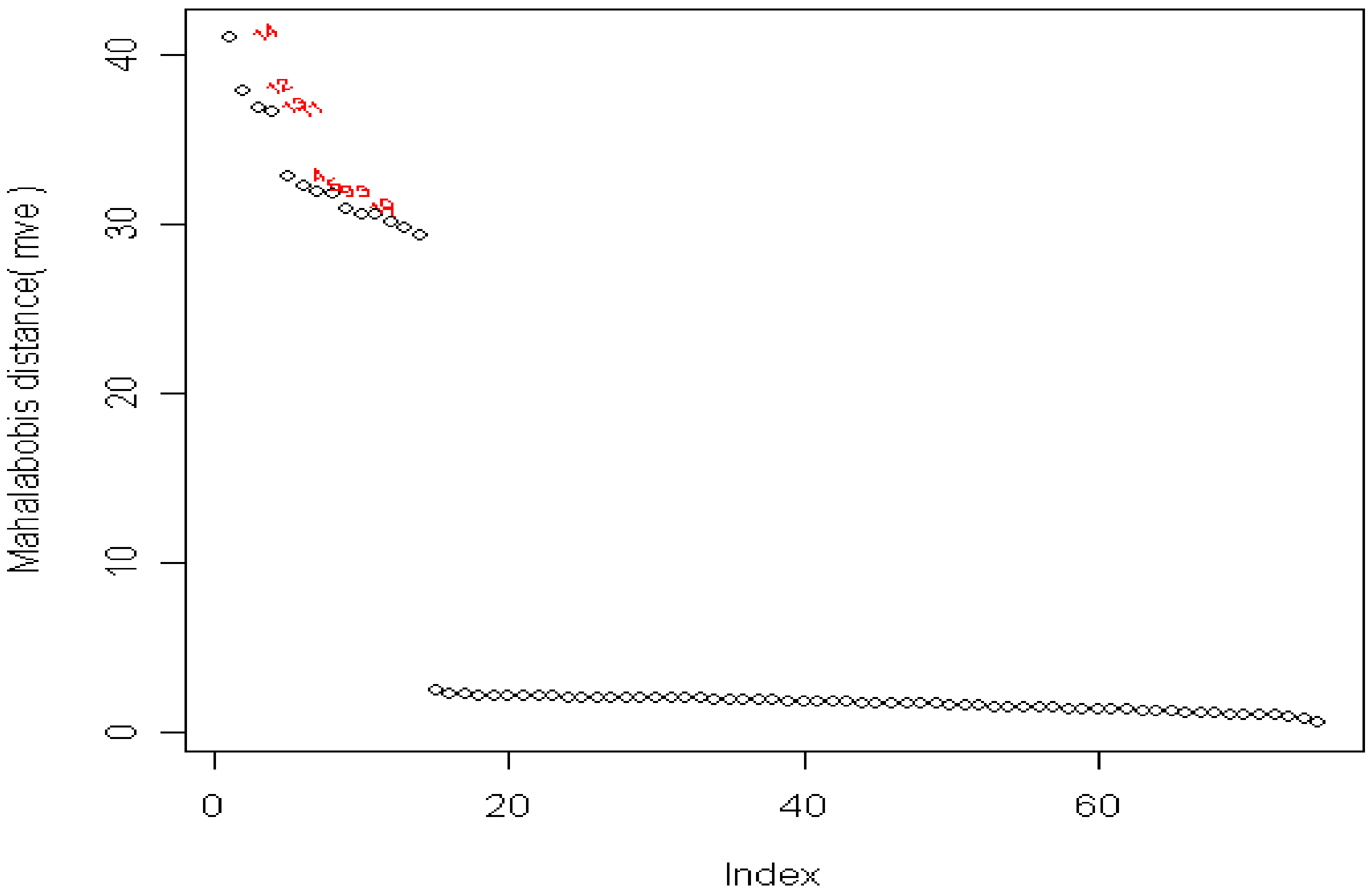
J={set of h instances: $Vol(S_J^*) \le Vol(S_K^*)$ for all K s. t. #(K)= h}.

$Vol(S_k)=\{|S_k| \text{med}_{i=1,2\ldots h}\, d_i^2\}^{1/2}$, d_i represents the Mahalanobis distance of the i-th instance in S_k

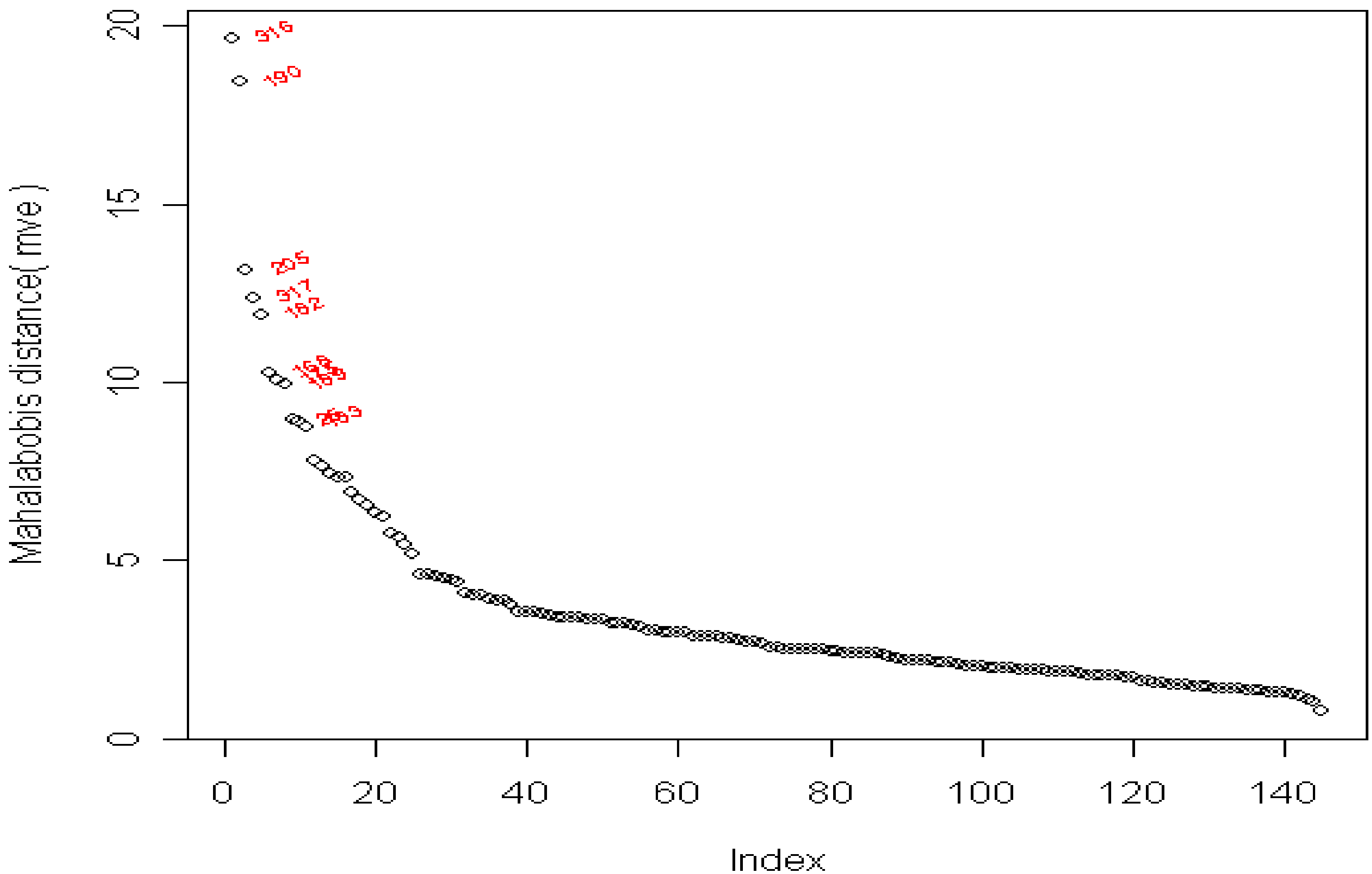
The elipsoid is defined by $(x-\bar{x})' S^{-1} (x-\bar{x}) \le a^2$.

The value of h can be thought of as the minimum number of instances which must not be outlying and usually h=[(n+p+1)/2], where [.] is the greatest integer function and p is the number of predictors.

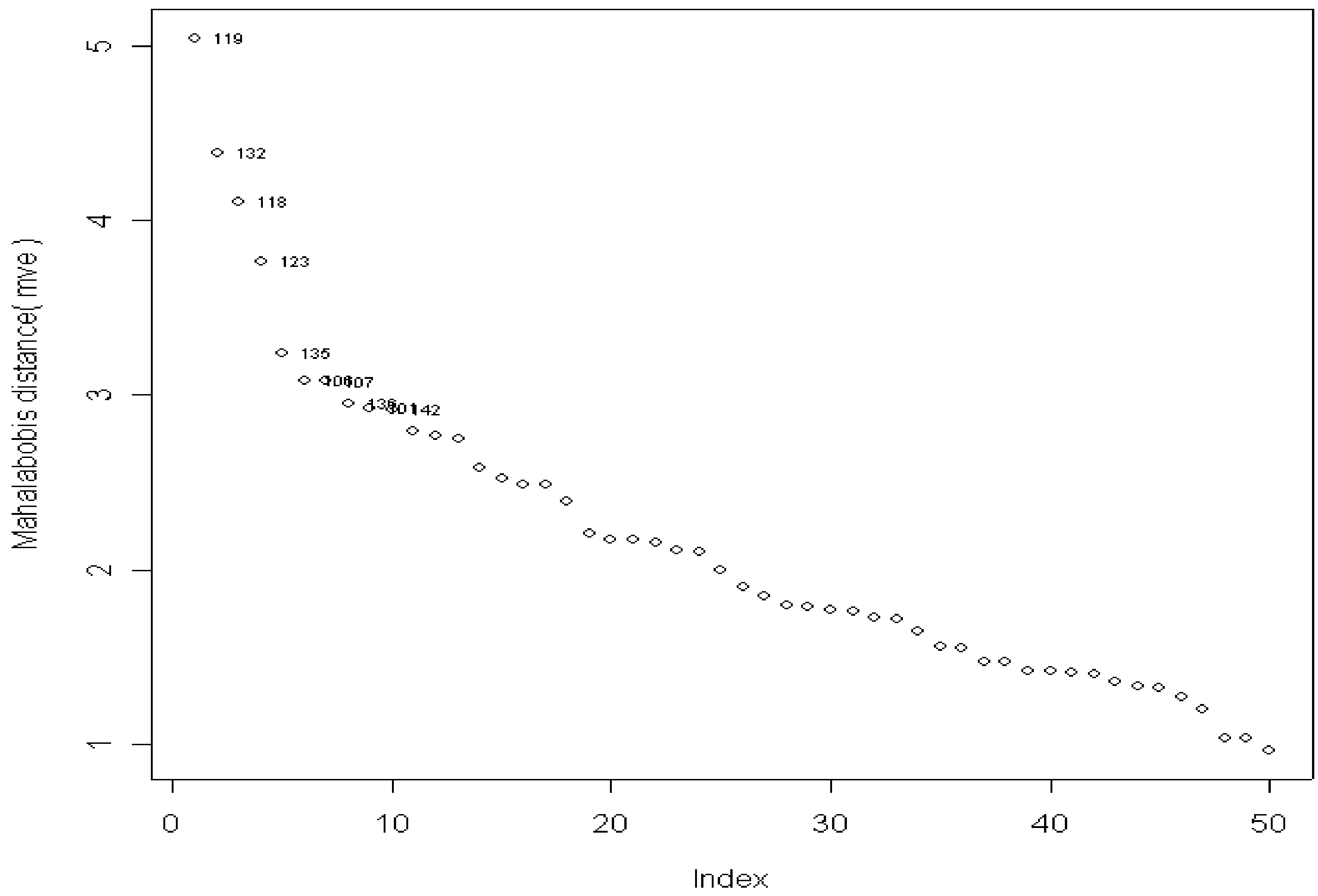
Outliers for the Hawkins dataset using MVE



```
 b=robout(bupa,1,"mve",10)

Top outliers by frequency

 25 148 167 168 175 182 183 189 190 205 261 311 312 316 317 326 343 345
 10  10  10  10  10  10  10  10  10  10   5  10  10  10  10  10  10  10

Top outliers by outlyngness measure
      316       190       205       317       182       168       189       175
20.886944 19.706658 13.804491 12.930697 12.690761 10.947778 10.713039
10.622882
       25       345       183       343       311       312       167       326
 9.316987  9.022665  8.915560  8.426077  8.156848  7.939317  7.667348
7.377310
      148       261
 7.305656  6.656579
```



Instances in the first class of Bupa ranked by theirs MVE estimators.



Plot of the instances in class 3 of Iris ranked by using a MVE estimator of theirs Mahalanobis distance

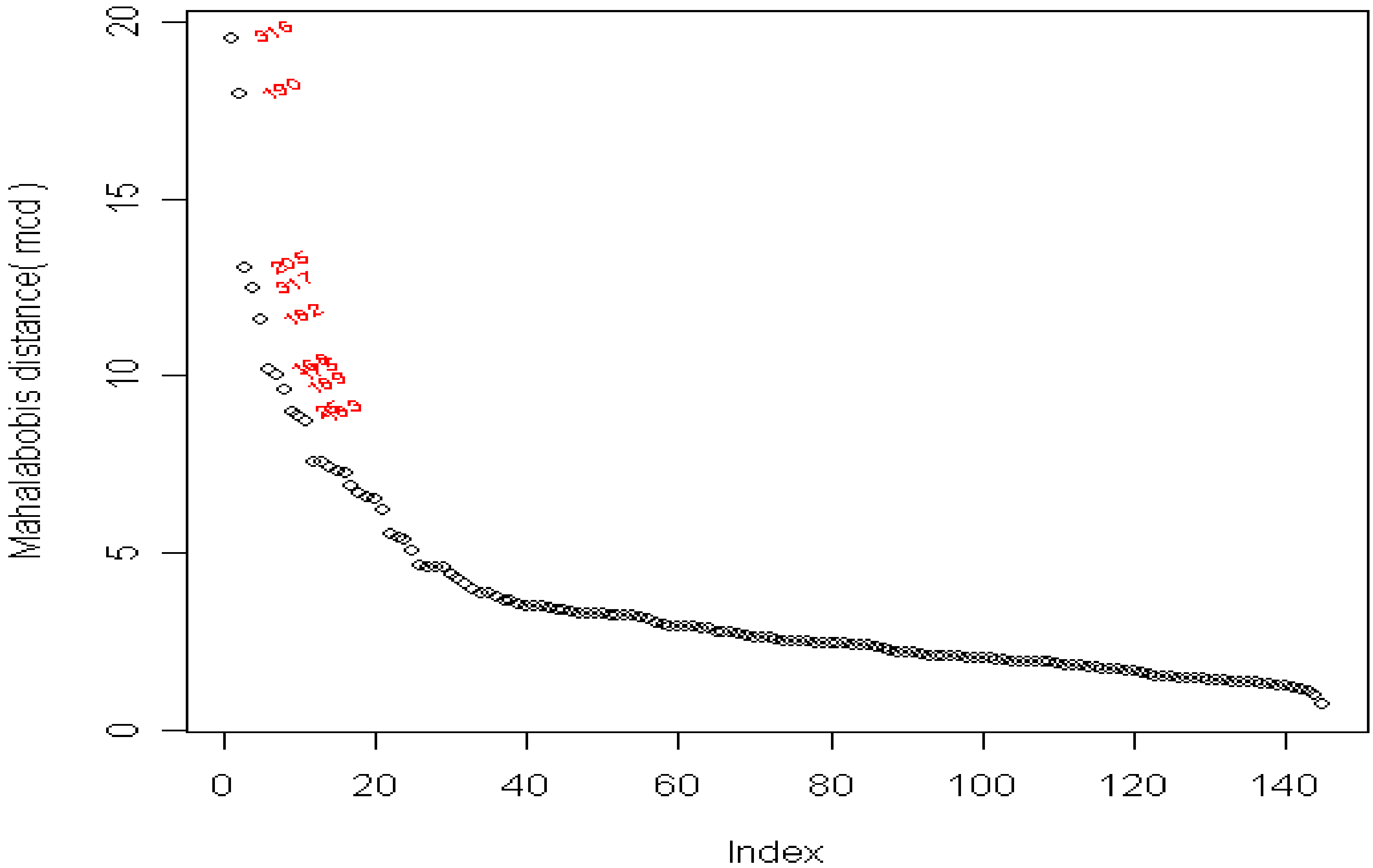
The ***Minimun Covariance Determinant (MCD)*** *estimator* is defined by

$$\text{MCD}=(\ \bar{\mathbf{x}}_J^*, S_J^*\ )$$

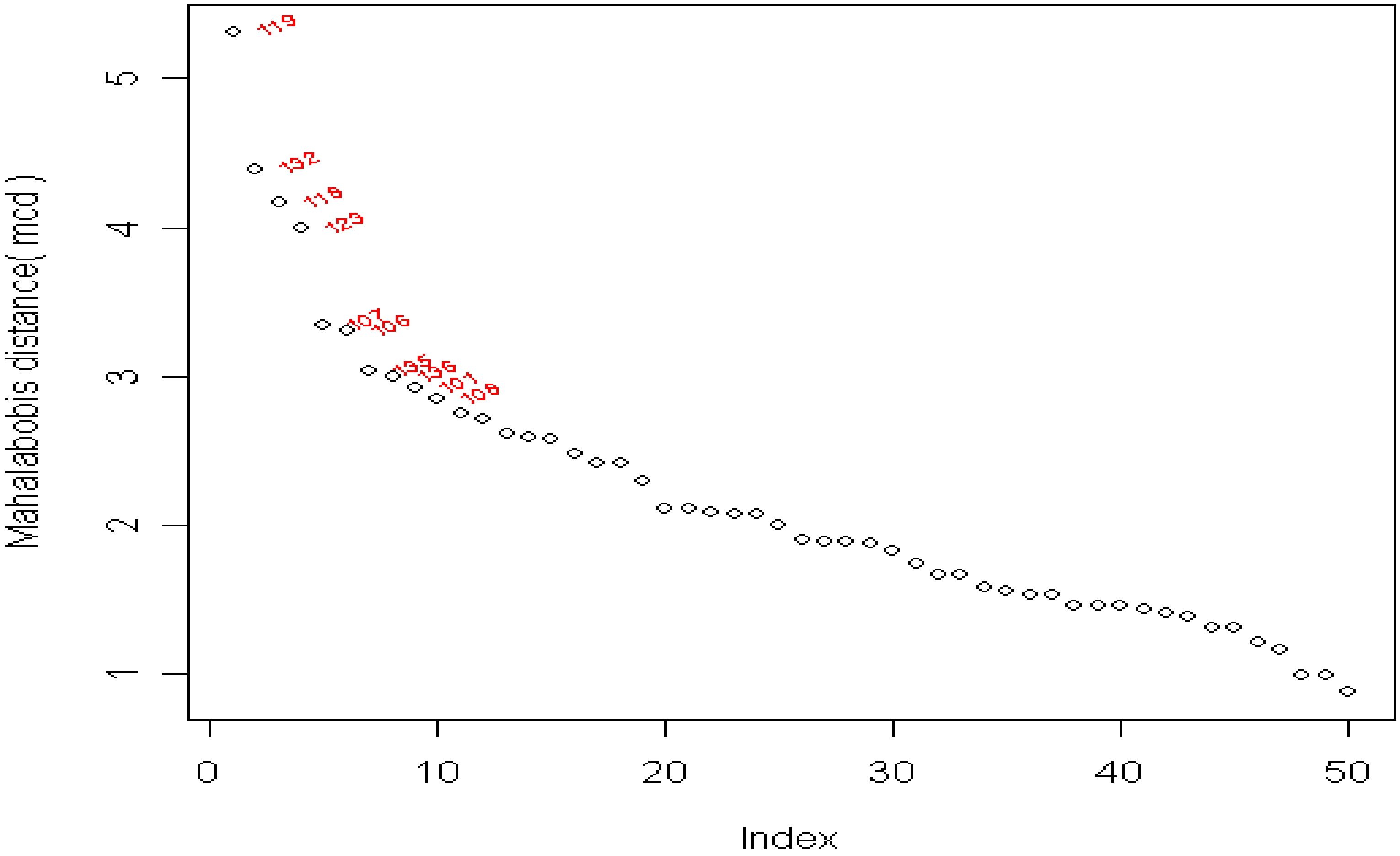
where J={set of h instances: $|S_J^*| \leq |S_K^*|$ for all K s. t. #(K)= h}.

H is defined as in the MVE estimator and |S| denotes the determinant of S

```
> b=robout(bupa,1,"mcd",10)
Loading required package: MASS

Top outliers by frequency

 25 148 167 168 175 182 183 189 190 205 261 311 312 316 317 326 335 343
344 345
 10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10   8  10

Top outliers by outlyngness measure
      316       190       205       317       182       168       175       189
19.462668 17.911479 12.984144 12.379654 11.566284 10.116878  9.958513
9.638781
       25       183       345       343       311       312       326       167
 8.907643  8.899936  8.735489  7.583311  7.533905  7.372262  7.342308
7.237911
      148       335       261       344
 6.863190  6.785321  6.558075  6.488697
```



Instances in the first class of Bupa ranked by theirs MCD estimators.



Plot of the instances in class 3 of Iris ranked by using a MCD estimator of theirs Mahalanobis distance

Top outliers per class in the Iris dataset by frequency and the outlyingness measure using the MCD estimator

Instance	Class	Frequency	Outlyingness
44	1	10	6.5574
24	1	10	5.9604
69	2	10	6.2246
119	3	10	5.3908
132	3	7	4.3935

Detection of outliers using clustering

Scattered outliers will form a cluster of size 1 and clusters of small size can be considered as clustered outliers. There are a large number of clustering techniques. Here, we only considered the Partitioning around Medoids (PAM) method. It was introduced by Kaufman and Rousseeuw (1990) uses k-clustering on medoids to identify clusters.

The function pam in library cluster of R performs clustering by PAM . Self-organizing maps SOM also can be used.

```
bupa1=bupa[bupa[,7]==1,1:6]
pambupa1=pam(bupa1,20,stand=T)
pambupa1$clusinfo
bupa1[pambupa1$clustering==16,]
```

```
> pambupa1$clusinfo
      size max_diss  av_diss diameter separation
 [1,]    4 2.831622 1.472387 4.132538  1.0782113
 [2,]   18 3.132019 1.287187 4.091354  0.7194979
 [3,]   13 2.765870 1.282699 4.140111  0.7194979
 [4,]    6 2.641598 1.340249 3.390138  1.1381321
 [5,]    4 3.000214 2.056078 4.393341  2.0068724
 [6,]    9 2.109355 1.098276 3.184331  0.4472404
 [7,]    5 2.494665 1.555043 3.579272  1.2655316
 [8,]   17 2.585376 1.217838 3.181508  0.8305449
 [9,]    8 2.892715 1.691034 3.975708  1.1988504
[10,]   14 1.943327 1.332506 2.993509  0.4472404
[11,]    5 3.054875 1.435943 3.747178  1.6822584
[12,]   15 2.560942 1.330603 3.622406  1.0895113
[13,]    9 2.836937 1.874549 4.649333  1.1942510
[14,]    6 2.799898 2.063445 4.365481  2.6737346
[15,]    2 3.272551 1.636275 3.272551  3.4565653
[16,]    4 3.668182 2.297649 4.864836  3.0504605
[17,]    1 0.000000 0.000000 0.000000  5.5076072
[18,]    3 3.255083 1.664524 4.145253  2.5271266
[19,]    1 0.000000 0.000000 0.000000  4.7535354
[20,]    1 0.000000 0.000000 0.000000  4.8177808
> bupa1[pambupa1$clustering==19,]
    V1 V2 V3 V4  V5 V6
316 99 86 58 42 203  6
```

Top outliers in the first class of the Bupa dataset detected by the PAM algorithm

Instance	Separation
190	5.507
317	4.817
316	4.753
182	3.456
205	3.456
335	3.255
189	3.050
345	3.050
343	3.050
312	3.050

Distance based outlier detection

Given a distance measure on a feature space, two different definitions of distance-based outliers are the following:

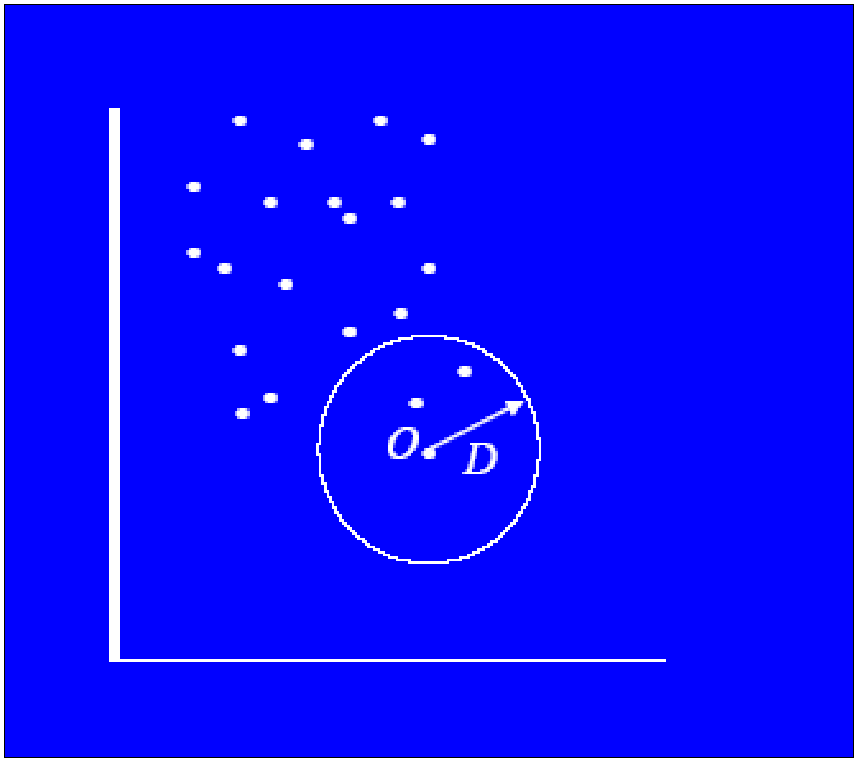
1. An instance **x** in a dataset D is an outlier with parameters p and λ if at least a fraction p of the objects are a distance greater than λ from **x**. (Knorr and Ng, 1997, 1998, Knorr et al. 2000). This definition has certain difficulties such as the determination of λ and the lack of a ranking for the outliers. Thus an instance with very few neighbors within a distance λ can be regarded as strong outlier as an instance with more neighbors within a distance λ.

2. Given the integer numbers k and n (k<n), outliers are the top n instances with the largest distance to their k-th nearest neighbor. (Ramaswamy et al., 2000). One shortcoming of this definition is that it only considers the distance to the k-th neighbor and ignores information about closer points. An alternative is to use the greatest average distance to the k nearest neighbors. The drawback of this alternative is that it takes longer to be calculated.

Distance-based outliers (cont.)

Formally, Object O in a dataset T is DB(p,D) outlier if at least a fraction p of the objects in T are at least distance > D from O.

e.g., DB(.99, 5) implies that 99% of data points are > 5 units distance away



Distance-based outliers (cont.)

- Bay and Schwabacher (2003) proposed a simple nested loop algorithm that tries to reconcile definitions 1 and 2, and at the same time reduce the theoretical time complexity of $O(kn^2)$ to almost linear in n, at least experimentally.
- The algorithm outputs *m* instances that have the greatest distance from their nearest *k* neighbors. The value of *m* is given by the user.

Bay's Algorithm

Input: k: number of nearest neighbors; n: number of outliers to return; D: dataset randomly ordered, BS: size of blocks in which D is divided.

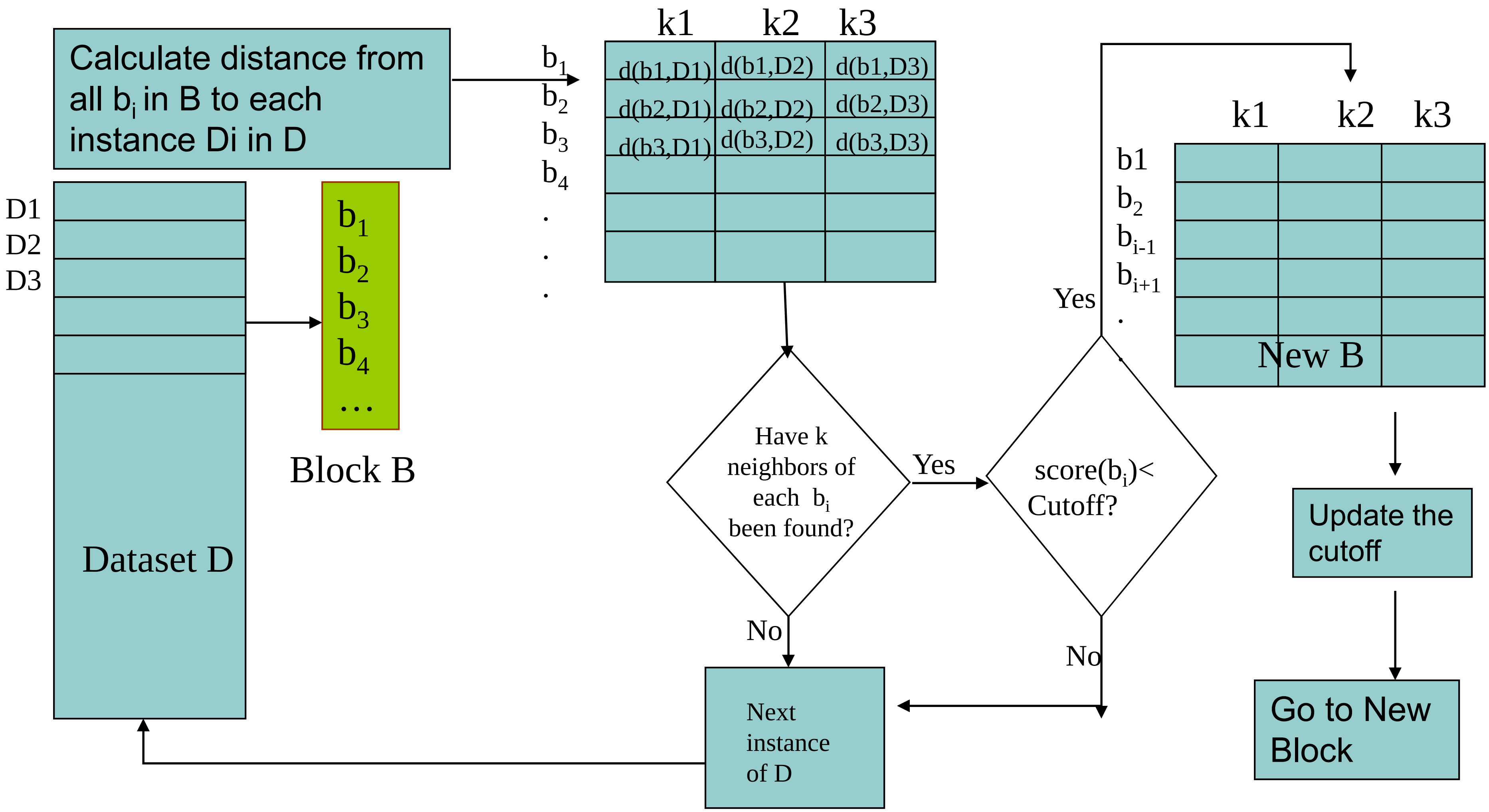
1. Let distance(x,y) return the Euclidean distance between x and y.
2. Let maxdist(x,Y) return the maximum distance between the instance x and the set of instances Y.
3. Let Closest(x,Y,k) return the k closest instances in Y to x.
4. Let score(Neighbors(x),x) returns median distance to the k neighbors of x.
5. Begin

Set the cutoff for c pruning to 0 and the set of outliers O as ϕ.

```
NB←ceiling(# instances in D/BS)
      While nb<NB {
       Neighbors(b)←ϕ for all b in block B_nb
       For each d in D {
             For each b in B_nb, b≠d{
                    If |Neigbors(b)|<k or distance(b,d)<maxdist(b,Neighbors(b)){
    Neighbors(b) ←Closest(b,Neighbors(b) ∪ d,k)
                 If(score(Neibghbors(b),b)<c){
                  Remove b from B_nb
                  }}}}
     O←Top(B(nb) ∪ O, n)  ; Keep only the top n outliers
     c←min(score(o)) for all in O ; The cutoff is the score of the weakest outlier
}
end
```

Output: O, a set of outliers

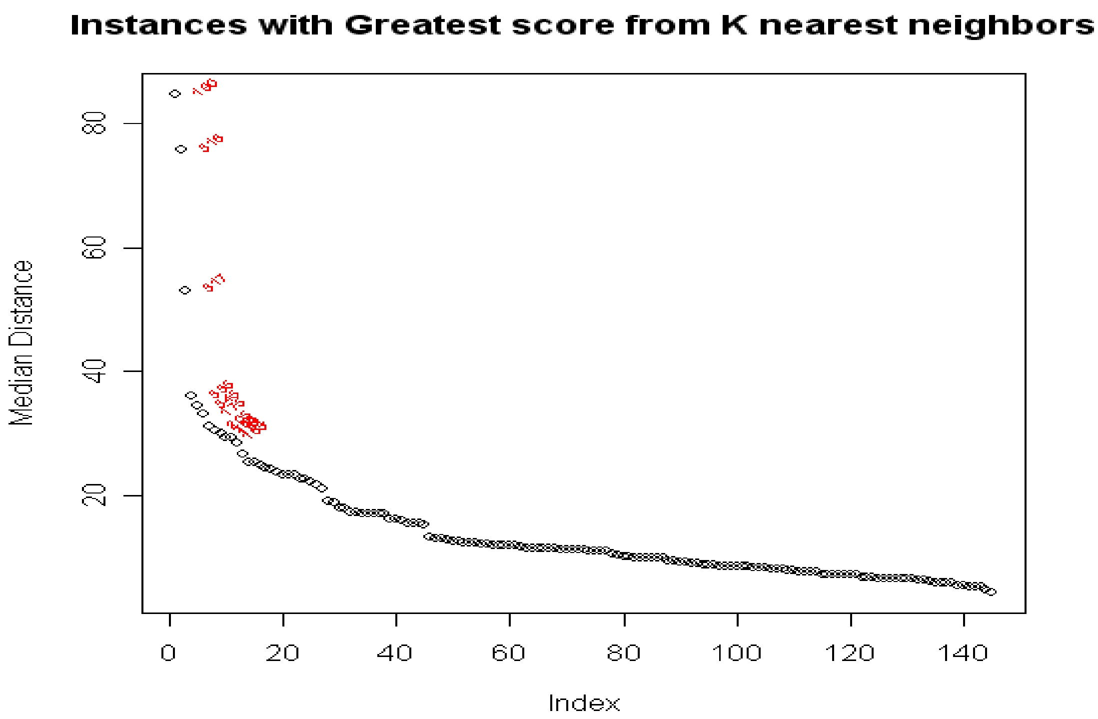
Bay's Algorithm – Nested loop with pruning rule

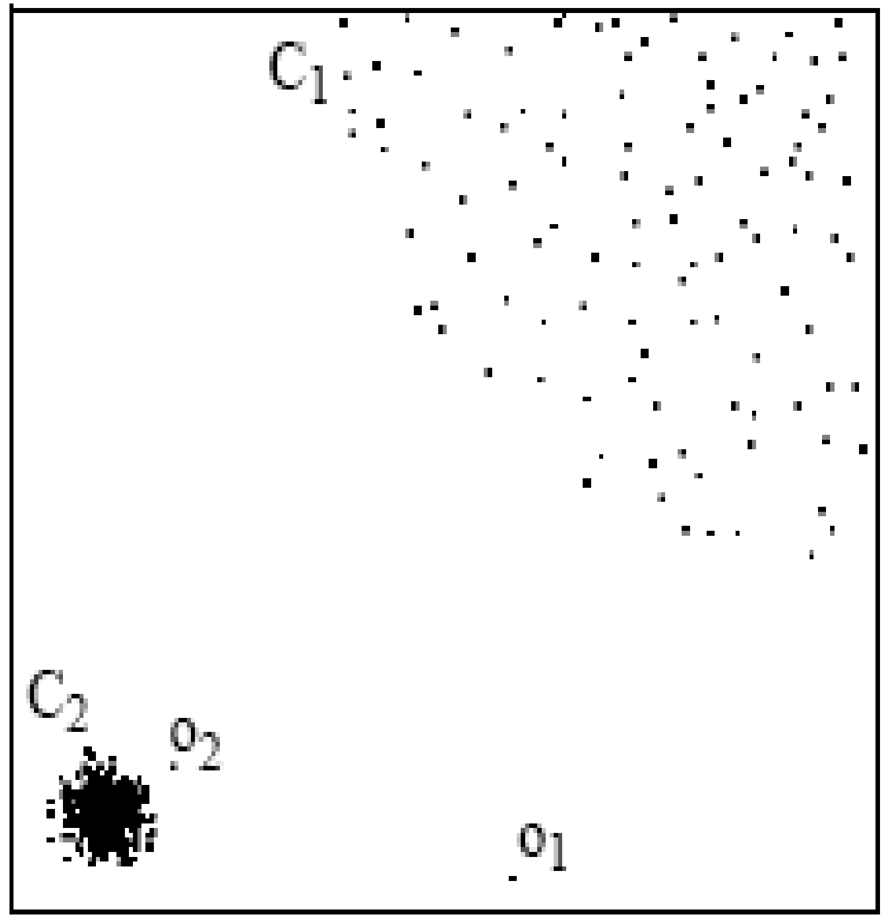


Outlier Detection using baysout( )

baysout(bupap[bupa[,7]= =1,1:6], blocks=10, num.out=10)

Index	Med. Dist.
190	84.700
316	75.810
317	53.103
335	35.961
345	34.481
175	33.075
205	31.228
168	30.483
182	30.016
183	29.325





A example showing the weakness of the distance-based method to detect outliers. O1 is detected but O2 is not detected.

Density-based local outliers

In this type of outliers the density of the neighbors of a given instance plays a key role (Breuning et al, 2000). Furthermore an instance is not explicitly classified as either outlier or non-outlier; instead for each instance a local outlier factor (LOF) is computed which will give an indication of how strongly an instance can be an outlier.

Definition 1. *k-distance of an instance x*

For any positive integer k, the k-distance of an instance x, denoted by k-distance(x), is defined as the distance d(x,y) between x and an instance y ε D such that:

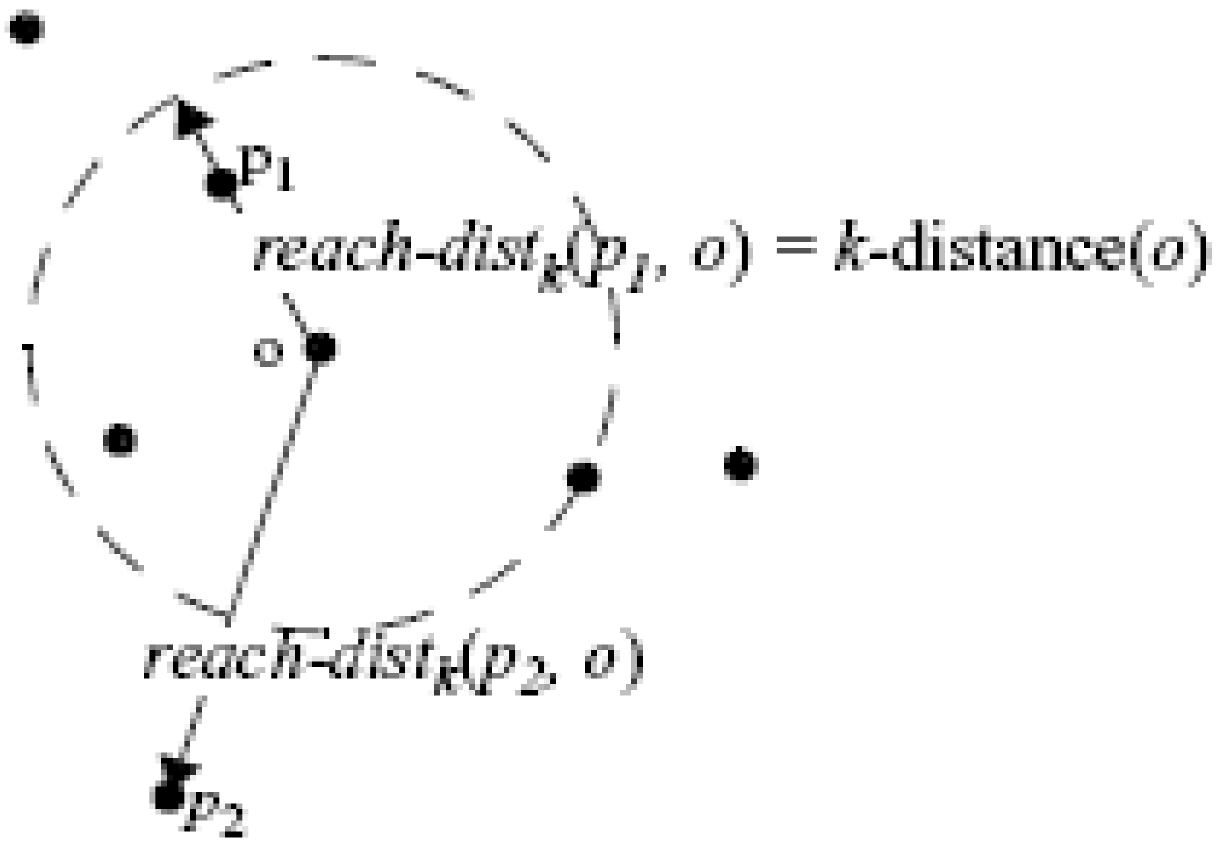
1. For at least k instances y' ε D-{x} it holds that d(x,y') $\leq$ d(x,y).
2. for at most k-1 instances y' ε D-{x} it holds that d(x,y') < d(x,y).

Definition 2: k-distance neighborhood of an object p
Given the k-distance of x, the k-distance neighborhood of x contains every object whose distance from x is not greater than the k-distance, i.e

$$N_{k\text{-distance}(x)}=\{y \in D-\{x\} \mid d(x,y) \leq k\text{-distance}(x)\}$$

These objects y are called the k-nearest neighbors of x.

Definition 3. *Reachability distance of an instance x w.r.t. instance y*
Let k be a positive integer number. The reachability distance of the instance x with respect to the instance y is defined as
$\text{reach-dist}_k(x,y)=\max\{k\text{-distance}(y),d(x,y)\}$



Reach-dist$_4$(p$_1$,o) y reach-dist$_4$(p$_2$,o) for k=4

The ***Local outlier factor (LOF)*** *of an instance x is defined by*

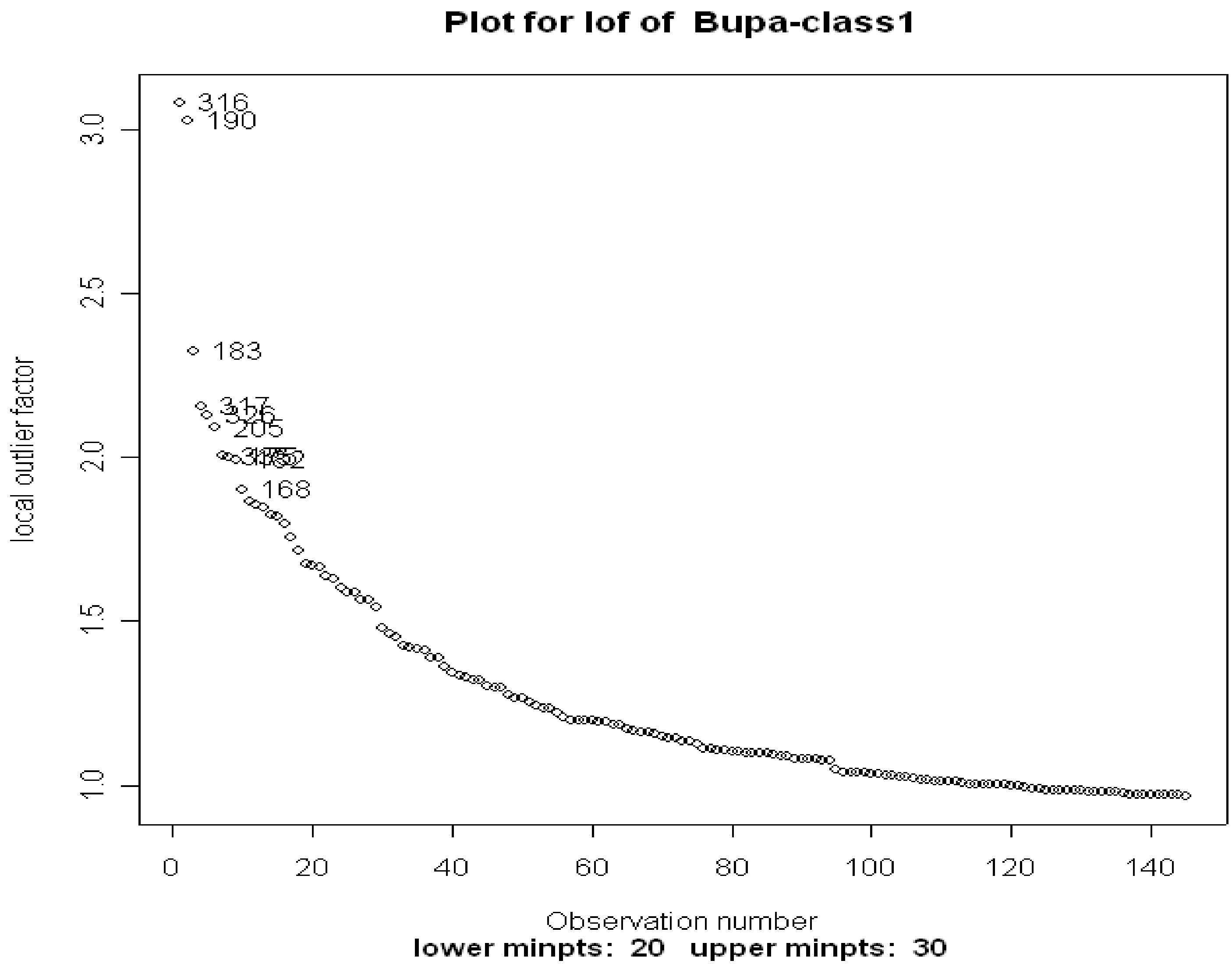
$$LOF_{MinPts}(x) = \left[\frac{\sum_{y \in N_{MinPts}(x)} \frac{lrd_{MinPts}(y)}{lrd_{MinPts}(x)}}{|N_{MinPts}(x)|} \right]^{-1}$$

where *lrd(.)* represents the *Local reachability density of an instance.* Given an instance x, its *lrd* is defined as the inverse of the average *reachability distance* based on the *MinPts*-nearest neighbor of the instance x.

Algorithm for detection of density-based local outliers

Input: Dataset D, MinptsLB, MinptsUB
Let Maxlofevct=ϕ
For each i in the interval [MinPtsLB, MinPtsUB]
{
1. Find the i nearest neighbors and their distance from each instance in D
2. Calculate the local reachability density for each instance in D
3. Compute the lof of each instance in D
4. Maxlofvect=max(maxlofvect, lof)
}
end
Output: Maxlofvect; the vector of maximum LOF's

```
> lofbupa1=maxlof(bupa[bupa[,7]==1,-7],"lofbupa1",20,30)
> lofbupa1[order(lofbupa1,decreasing=T)][1:10]
     316      190      183      317      326      205      335      175
3.079764 3.026225 2.324484 2.156292 2.128578 2.089691 2.003301 2.001225
     182      168
1.992892 1.903069
>
```



Plot of the instances in class 1 of Bupa ranked according to the LOF outlyingness measure.

Summary of outliers in Bupa class 1

Method	Outliers
Mahalanobis	190,316,317,345,183,335,205
MVE	316,190,205,317,182,168,189,175,25,345,183,343,311,312,167,326,148,261
MCD	316,190,205,317,182,168,189,175,25,345,183,343,311,312,167,326,148,261,344,335
PAM	190,317,316,182,205,335,345,312
Bay's	190,316,317,205,175,168,182,189,25,167
LOF	316, 190, 183, 317, 326, 205 , 335, 175, 182, 168

Evaluating the effect of outliers

Two main aspects to consider in supervised classification are the estimation of the misclassification error rate and feature selection.

Three classifiers considered: LDA, KNN and Rpart (a decision tree classifier)

Two feature selection methods: SFS(wrapper) and Relief(filter).

Error estimation method: 10-fold cross validation

Datasets: Iris.

The misclassification error rate for the LDA, knn and rpart classifiers in *Iris* using three different types of samples

Classifier	Original Sample	Deleting outliers	Deleting a random subsample
LDA	2.02	1.54	2.30
KNN(k=1)	4.05	2.35	4.10
Rpart	6.69	2.90	7.32

Features selected in *Iris* using SFS and Relief for the three type of samples

FS Method	Original Sample	Deleting Outliers	Deleting a random subsample
SFS(LDA)	4,2	4,2	4,3
SFS(KNN)	4,3	4,3	4,3
SFS(Rpart)	4	4	4
Relief	2,3,4	4,3	4,3

The misclassification error rate for the classifiers after feature selection in *Iris* using three different types of samples

Classifier	Original Sample	Deleting outliers	Deleting a random subsample
LDA	3.70	2.33	5.31
KNN(k=1)	4.01	1.87	4.80
Rpart	5.29	2.29	5.25

Some conclusions

- There is not a unique method to detect all outliers.
- There is not a major effect on the feature selection method. Relief seems to be more affected.
- LDA and KNN classifiers seem to be more affected by the outliers than the Rpart.
- Use Visualization