Some Approaches to Outliers' Detection in R

Marcello D'Orazio (madorazi@istat.it)
Italian National Institute of Statistics - Istat

ABSTRACT

Outlier detection is part of data editing phase for numerical variables. This work investigates outlier detection in the R environment by comparing "traditional" methods, popular in official statistics, with techniques developed in the field of data mining and statistical learning. The comparison is done considering longitudinal data where a set of quantitative non-negative variables are observed twice (or more) on the same sample of units. The work attempts to identify some "recent" outlier detection methods, already available in the R environment, that seem suitable for application in official statistics. This study takes stock of findings of a previous work investigating outlier detection in the univariate case that showed the goodness of some "recent" approaches; in this article we go a step further and investigate the behavior of "traditional" and recent methods also in the multivariate case. The first preliminary results are quite interesting and useful as guidance towards application of the chosen methods in the production of official statistics using the R facilities.

Keywords: *binary recursive partitioning, clustering, nearest neighbor distance, panel data.*

JEL classification*: C – Mathematical and Quantitative methods; C14 Semiparametric and Nonparametric Methods: General; C33 Panel Data Models; C83 Survey Methods*

1. INTRODUCTION

National Statistical Institutes (NSIs) spend non-negligible efforts in checking the incoming data to detect actual or potential errors. This *data editing* phase (or *statistical data editing*, sometimes also referred as to *input data validation*) can make use of a variety of statistical methods depending on the type of variables (continuous, categorical, or mixed-type) and the relationships existing between them; the causes of errors and how they can affect the final estimates; the data collection mode; etc. This note concentrates on the subset of data editing methods tailored to *outlier detection*; according to UNECE (2000) "an outlier is a data value that lies in the tail of the statistical distribution of a set of data values", whereas "outliers in the distribution of uncorrected (raw) data are more likely to be incorrect". Waal *et al* (2011, pp. 7-8) give a more general definition: "a value, or a record, is called an outlier if it is not fitted well by a model that is posited for

the observed data"; it is said *univariate outlier* if it is a single value of the whole record; while, on the contrary, it is called *multivariate outlier* if it consists of "an entire record, or at least a subset consisting of several values, is an outlier when the values are considered simultaneously, that is if they do not fit the posited model well when considered simultaneously".

When observing a single continuous variable (household income; firm production; harvested area in a farm etc.) an outlier is often caused by a *measurement error*, i.e. an error occurred in data collection and such that the observed value is not equal to the true value (and the true value is not expected to be in the tail of the distribution), in this case the outlier should be removed and replaced (imputed). In some cases, an outlier can also be a non-erroneous "extreme" value which, however, may have a great influence on the final estimates (*influential value*) and therefore may deserve a "special" treatment in the analysis. Waal *et al* (2011, p. 8) note that an influential value "is often an outlier, and vice versa; however, an outlier may also be a noninfluential value and an influential value may also be a nonoutlying value". For this reason, detection of outlies often takes place in *selective-editing* sub-phase tailored to identify outliers and *influential errors*.

This work investigates traditional and "recent" approaches to outlier detection currently implemented in the R environment (R Core Team, 2022) by carrying out a small empirical comparison on panel survey data, where a continuous variable is observed on the same set of units (households, firms or agriculture holdings) in different time occasions. Next Section briefly describes well-known outlier detection methods based on fitting explicit statistical models; in addition, it summarizes the Hidiroglou and Berthelot (1986) method, developed for outlier detection with panel survey data. Section 3 introduces "traditional" and "recent" nonparametric outlier detection methods, mainly proposed in the field of data mining or machine learning, which seem suitable for application in official statistics. Since many of these recent methods rely on calculation of distance between observation, Section 4 discusses the major issues related to application of distance-based approaches in the multivariate case. Section 5 compares the results provided by the chosen methods with data related to farms and firms. Finally, Section 6 summarizes the main findings and future areas of work.

2. PARAMETRIC APPROACHES TO OUTLIER DETECTION

This Section is not intended to provide a comprehensive overview of parametric methods available for detecting outliers, rather it just mentions some popular approaches implemented in the R environment.

Formally, in the univariate case we consider a single quantitative non-negative variable *Y* observed on a random sample of *n* units ($y_i \geq 0;\ i = 1,2, \dots, n$). In a parametric framework, it is common to assume a Gaussian distribution (for raw data or log-transformed raw data) and then search for outlier in the tails of the estimated distribution; robust methods are applied to estimate the location and scale parameters; the median is a popular robust estimate of the location parameter, while several alternative robust estimators of scale parameter exist, ranging from simple ones considering the *inter quartile range* (IQR), to more complex ones using the *median absolute deviation* (MAD), S_n and Q_n estimators (see Rousseeuw and Croux, 1993) and τ estimator proposed by Maronna and Zamar (2002). In R all these robust estimators are made available by the package **robustbase** (Maechler *et al*., 2022). A wrapper to most of the methods is included in the package **univOutl** (D'Orazio, 2022).

A different approach tries to identify outliers by assuming that data are generated by a mixture of two Gaussian distributions sharing the same location but with different scale. The underlying idea is that the "contaminated" data points (outliers) are caused by additive measurement errors with zero mean but variance proportional to that of non-contaminated data. In this setting, estimation of parameters of the two distributions allows to identify the outliers as those observations that have higher probability of being generated by the contaminated Gaussian distribution (see e.g. Di Zio and Guarnera, 2013). This approach is implemented in the R package **SeleMix** (Guarnera and Buglielli, 2020) that allows including error-free predictors of *Y*.

When data do not follow the Gaussian distribution, it is possible to resort to approaches based on other models; for instance, the R the package **extremeValues** (van der Loo, 2010) covers Gaussian and Log-Normal as well as Weibull and Pareto distributions (van del Loo, 2010). Alfons *et al*. (2013) suggest to fit a semi-parametric Pareto tail model to detect outliers in complex sample surveys designed to estimate indicators on social exclusion and poverty (in EU traditionally produced by the European Union statistics on income and living conditions survey); this approach is implemented in the R package **laeken** (Alfons and Templ, 2013).

In the multivariate case we consider *p* continuous variables, $\mathbf{y}_i' = (y_{i1}, \dots, y_{ip})$ having non-negative values ($y_{ic} \geq 0;\ i = 1,2, \dots, n; c = 1, \dots, p$) observed on a sample of *n* units. When assuming a multivariate Gaussian distribution a wide set of methods identify potential outliers as observations having the largest Mahalanobis distance from the center of the data. In this setting the squared Mahalanobis distance follows a Chi-Square distribution with *p* degrees of freedom and it is common

to identify as potential outliers those units having a squared distance greater than $\chi^2_{p,1-\alpha}$. Commonly adopted methods to achieve robust estimates of the parameters of the multivariate Gaussian distribution are MCD, MVE, OGK, etc. (see e.g. Todorov and Filzmoser, 2009). The robust estimation of location and variance-covariance parameters allows to compute a robust Mahalanobis distance and, accordingly, observations whose robust distance is greater than $\chi^2_{p,1-\alpha}$ are identified as potential outliers. This way of working basically identifies α percent of observations as outliers; for this reason, Filzmoser *et al.* (2005) suggest an "adaptive" approach that compares the theoretical distribution (χ^2_p) with the empirical distribution of the squared robust distances. All these features are implemented in the R package **mvoutlier** (Filzmoser and Gschwandtner, 2021; in particular see functions `arw()` and `dd.plot()` that are based on MCD estimator for the parameters of the multivariate Gaussian distribution). It is worth noting that the package **mvoutlier** includes various methods for multivariate outlier detection (see Filzmoser and Gschwandtner, 2021).

The package **SeleMix** (Guarnera and Buglielli, 2020) permits to apply the mixture-based approach also in the multivariate case.

A special multivariate case occurs when one or more continuous non-negative variables are observed repeatedly over time on the same set of units (panel surveys); in this case it is expected a high correlation between subsequent measurements and this feature becomes crucial when the objective is the estimation of the change over time of a population parameter related to one of more of the considered variables. In the bivariate case ($p = 2$) a very popular approach is suggested by Hidiroglou and Berthelot (1986).

2.1 Hidiroglou-Berthelot method for outlier detection with longitudinal data

Hidiroglou and Berthelot (1986) suggest to detect outliers by analyzing scores obtained by transforming the ratios $r_i = y_{t_2 i}/y_{t_1 i}\,(i = 1,2, \ldots, n)$; r_i denotes the "individual change" from time t_1 to time t_2 ($t_2 > t_1$) for unit *i*. In practice, at first ratios are transformed in the following manner:

$$s_i = \begin{cases} 1 - \frac{r_M}{r_i}, & \text{if} \quad 0 < r_i < r_M \\ \frac{r_i}{r_M} - 1, & \text{if} \quad r_i \geq r_M \end{cases} \qquad [1]$$

where r_M is the median of the ratios, then to account for the magnitude of data and give more "importance" to units involving high values of *Y*, the following *scores* are derived:

$$E_i = s_i\left[\max\left(y_{t_1 i}, y_{t_2 i}\right)\right]^U \quad [2]$$

where U can range from 0 to 1 ($0 \leq U \leq 1$) and controls the role of magnitude in determining importance associated to transformed ratios (a common choice consist in setting $U = 0.5$). Finally, assuming that E-scores follow a Gaussian distribution the potential outliers are the units whose scores fall outside the interval:

$$\left[E_M - C \times f_{Q1}, E_M + C \times f_{Q3}\right] \quad [3]$$

Being E_M the median of the E scores, while f_{Q1} and f_{Q3} are functions of the quartiles of the E-scores with a correction factor that avoids drawbacks of distributions highly concentrated around the median; in addition, the bounds allow for a slight skewness in the distribution of the E scores. Recently, Hidiroglou and Emond (2018) suggest to replace the quartiles with the deciles ($P_{10,E}$ and $P_{90,E}$ instead of respectively $Q_{1,E}$ and $Q_{3,E}$) in cases where a large proportion of units (>1/4) share the same value of the ratio, since in this case the "standard" method would detect too many observations as potential outliers. The parameter C in expression [3] determines how far from the median the bounds should be; commonly suggested values are $C = 4$ or $C = 7$ but larger values can be considered, depending on the tails of the distribution of the E scores. Practically, the choice of the constants U and C is not straightforward and it is preferable to graphically investigate how the E scores are distributed.

It is worth noting that data editing literature suggests alternative methods to check whether the individual change (r_i) is too large or too low (see e.g. the theme "Editing for Longitudinal Data" in Eurostat, 2014).

In the R environment the Hidiroglou Berthelot (HB) procedure is implemented in the package **univOutl** (D'Orazio, 2022) that includes also graphical facilities for inspecting the scores, in line with Hidiroglou and Emond (2018) suggestion.

3. NONPARAMETRIC APPROACHES TO OUTLIER DETECTION

Nonparametric outlier detection methods avoid explicit assumption on the underlying distribution; in this group fall many outlier detection methods proposed in the domain of data mining and statistical learning. This Section summarizes the features of a subset of popular methods that have the advan-

tage of being easily applicable in the production of official statistics by using the facilities of the R environment. D'Orazio (2023) carried out an empirical investigation of some of these methods but only in the univariate setting.

3.1 Boxplot

Detecting outliers by plotting a *boxplot* (*box-and-whisker* plot) is very popular in the univariate case; the units outside the *whiskers* are considered outliers. To account for skewness Hubert and Vandervieren (2008) suggested an "adjusted" boxplot taking into account a measure of the skewness, the *medcouple*, that works with moderate skewness. The R package **univOutl** (D'Orazio, 2022) includes a series of functions for outlier detection based on the standard or adjusted boxplot.

3.2 Outlier detection based on nonparametric density estimation

Literature on data mining and statistical learning provides many suggestions for outlier detection, often indicated as *unsupervised* outlier detection methods; the largest group of methods is the one "inspired" to nonparametric estimation of density. As noted by Zimek and Filzmoser (2018), several approaches under this umbrella simply consists in calculating distances between observations. Some of them are variants of Ramaswamy *et al.* (2000) proposal that suggest identifying the potential outliers by calculating the *k* nearest neighbor (*k*-NN) distance; in practice, if $d_{i,(k)}$ is the distance of the *i*th from its *k* nearest neighbor, then units showing largest values of $d_{i,(k)}$ are potential outliers. Angiulli and Pizzuti (2002) suggest to analyze the "weight" obtained by summing up all the distances from the corresponding *k* nearest neighbor observations:

$$\omega_{i,(k)} = \sum_{j=1}^{k} d_{i,(k)} \qquad [4]$$

Similarly, Hautamäki *et al* (2004) suggest using the average of distances ($\overline{\omega}_{i,(k)} = \omega_{i,(k)}/k$). Campos et al. (2016) note that the sum (or average) of distances reduces the variability of the scores and return scores less sensitive to the value of *k*.

In general, there's no rule-of-thumb for deciding *k*; D'Orazio (2023) investigated these approaches in the univariate case highlighting the difficulties in analyzing the final scores (distance or "weight") whereas the magnitude, indicating the chance of being an outlier, increases by increasing the value of *k*. In particular, it seems difficult to identify a threshold such that scores greater than it can be considered potential outliers. Hautamäki *et al.* (2004) suggest to derive the threshold as a fraction of the observed maxi-

mum first order difference calculated after sorting the scores in increasing order. Practically a graphical inspection can be more effective: after sorting the scores increasingly, good candidate thresholds are the values corresponding to "jumps" in the plot (abnormal increase in the score). In this setting, a good candidate threshold is the point of maximum curvature in the graph, as shown later (see Section 3.3).

The before mentioned approaches are labeled as "global" in data mining literature and for this reason not flexible enough to catch situations where sub-groups of observations may have different "local" densities; this latter case is better handled by approaches developed starting from the *local outlier factor* (LOF) (Breuning *et al*, 2000). The idea is very simple and consists in comparing the *local reachability density* of each observation with the average local reachability calculated on the q nearest neighbors (the *local reachability density* is obtained as the inverse of the average *reachability distance* between each unit and its q closest neighbors). An outlier is expected to have a local reachability distance smaller than the average on neighboring units and consequently a LOF score greater than one. The larger is the LOF score the higher is the chance of finding an outlier; unfortunately, also in this case there's not a rule for setting a threshold.

The R package **DDoutlier** (Madsen, 2018) provides functions to apply many density and distance-based outlier detections approaches. *k*-NN distance is however calculated in many other R packages, the main drawback is that most of them permit to use just a limited set of popular distance functions (typically Euclidean and Manhattan distances).

3.3 Outlier detection with clustering-based algorithms

Density-based spatial clustering with noise (DBSCAN; Ester *et al*, 1996) separates the observations that do no not belong to any cluster, because not "reachable" by any other observations, as "noisy" observations, i.e. outliers. The "reachability" depends on a distance threshold ε; in practice two units *i* and *j* are *directly reachable* if their distance is less or equal than ε ($d_{i,j} \leq \varepsilon$), while they are only *reachable* if there is a path of three of more observations to go from *i* to *j*, where each couple of units in the path *is directly reachable*. In addition, the DBSCAN algorithm requires setting also the number g of "core" observations, i.e. observations that have at least $g - 1$ distinct units at a distance smaller or equal to ε. The literature suggests to set $g = 2p'$ (cf. Schubert *et al*, 2017) where p' ($p' \leq p$) is the number of variables used to calculate the distance, i.e. those for which we search for potential multivariate outliers. Empirical results in Schubert *et al* (2017) seem to support the conclusion that often g has a limited impact on the results; on the contrary ε is cru-

cial. A common suggestion is to plot the $(g-1)$-NN distances in increasing order and set ε equal to the distance where the plot shows a "valley," "knee," or "elbow" (Schubert *et al*, 2017). Unfortunately, with many observations the graphical representation may show points rather close each other and difficult to investigate. A simpler criterion could be that of approximating ε with the point of maximum curvature, i.e. the point where the empirical curve has the maximum distance from the straight line connecting the smaller and the larger $(g-1)$-NN distances. This is basically the approach for identifying a "knee" in a series of discrete points showing an increasing concave-down curvature (plot of sorted $(g-1)$-NN distances usually shows an increasing concave-up curve); in particular, in this paper we opt for the *Kneedle algorithm* (Satopaa *et al.*, 2010) that is simple and can be easily implemented in R.

Although there are several proposals to improve the seminal DBSCAN idea, it still remains an approach that provides quite good results. In R the DBSCAN is made available by the package **dbscan** (Hahsler *et al.*, 2019; Hahsler and Piekenbrock 2022).

3.4 Outlier detection with recursive partitioning trees

The underlying idea is that outliers have a higher chance of being separated by the other ones in one branch of the partitioning tree with relatively few splits. In the univariate case an arbitrary threshold y_o is selected at random within the range of *Y* ($y_c^{(min)}$, $y_c^{(max)}$) and all the observations are divided into two groups according to whether they show higher or lower values than y_o. This randomized splitting process is applied recursively (i.e., divide the units into two groups then repeat the process in each group, and so on) until no further split is possible (or until meeting some other criteria). The final outcome is an *isolation tree* where the more observations show similar *Y* values, the longer (more splits) it will take to separate them in small groups (or alone) compared to less occurring *Y* values; for this reason, the *isolation depth* (number of splits needed to isolate a unit) can be considered as a tool for detecting outliers.

Since the isolation depth estimated in a single isolation tree would be characterized by a high variability, the common adopted device consists in building an ensemble of isolation trees – the *isolation forest* – and then derive the final score by averaging over the fitted trees (Liu *et al.*, 2008 and 2012). Like random forests, each single isolation tree can be fit on a bootstrap sample of *m* ($m < n$) observations randomly selected. In the Liu *et al* proposal (2008 and 2012) the partitioning stops when a node has only one observation or all units in a node have the same values (in some cases it is introduced a maximum value for the tree height). The isolation forest returns a score u_i ranging

from 0 to 1 ($0 < u_i \leq 1$); scores close to 1 indicate observations with a very short average path length that tend to be isolated earlier than the other ones and therefore denote outlying observations. As a consequence, setting a threshold u_0 ($0 < u_0 < 1$), returns as outliers all the units having a score $u_i > u_0$; it is suggested to consider $u_0 = 0.5$ but D'Orazio (2023) shows that a graphical inspection of the ordered scores can be beneficial in deciding u_0.

The isolation forest is very efficient and requires setting just two tuning parameters, the subsample size *m* and the number of trees to fit. Liu *et al* (2008 and 2012) claim that even a small subsample size ($m = 256$) can work with very large data-sets, even though some implementations of the algorithm avoid subsampling if *n* is not very large. Concerning the number of trees to fit, it is suggested to start with at least 100, but this figure should be increased when the achieved scores are on average quite below 0.5, as this may point out a problem of unreliable estimation of the average path length.

In the multidimensional framework, the various trees are derived by picking at random at each iteration one of the available variables. Unfortunately, the standard way of working (branching and bounding) would consist in an ensemble of results related to the application of isolation forest independently variable by variable. To compensate this drawback, it is preferable to consider an *extended isolation forest* (Hariri et al 2018) that in the branching step considers jointly two or more variables; for instance, with $o = 2$ variables the algorithm partitions repeatedly the units according to a regression line whose intercept and slope are randomly generated each time (when $o > 2$ the branching considers randomly generated hyperplanes). Liu *et al* (2010) suggest to set $o = 2$ as it seems to work well even with many starting variables. Hariri *et al* (2018) suggestion is to set $o = 2$ or $o = 3$.

In R the standard isolation forest is implemented in the package **solitude** (Srikanth, 2021) while the package **isotree** (Cortes, 2022) permits to fit also the extended isolation forest.

4. ISSUES IN COMPUTING DISTANCES IN THE MULTIVARIATE FRAMEWORK

Many of the outlier detection methods presented in the previous sections rely on calculating distances between observations. This is often perceived as a very simple way of working, although in the multidimensional setting it involves taking additional decisions on: (i) the distance function, and (ii) whether the variables need to be scaled in advance. These choices are often understated and the practitioner accepts without criticism the default choices made by the developers of the used software packages, without

worrying whether the default settings can work in their case studies. A common choice of many distance-based outlier detection is that of considering the Euclidean distance or the Manhattan distance, that are particular cases of the *L*-norm :

$$d_{i,j} = \sqrt[T]{\sum_{c=1}^{p} |y_{ic} - y_{jc}|^T} \quad [5]$$

with respectively $T = 2$ and $T = 1$. This expression shows that a variable having larger values than another tends to dominate the overall distance and the dominance increases with increasing values of *T*. In multivariate outlier detection this would mean that an outlier in a distance-dominant variable influences greatly the detection of multivariate outliers; for this reason, before calculating the distances it may be necessary to scale the variables (preliminary scaling is not needed when applying the Mahalanobis distance).

Common choices for scaling the variables are the range or the standard deviation but, as in the case of outlier detection with Gaussian distributed data, the scaling should involve a robust estimate of the standard deviation or replacing the range with a function of inter-percentile range (e.g. difference between the whiskers of the boxplot, etc.).

Another common mistake in multivariate outlier detection is to jointly consider all or almost all the *p* available numerical variables despite how large is *p*. With a quite high number of continuous variables there is the risk of incurring in effects of the *curse of dimensionality*. In particular, in unsupervised outlier detection with a high-dimensional dataset, Zimek *et al* (2012) stress that the major problem of distance-based methods is the loss in discrimination ability, i.e. a reduction of the ability of the distance in discriminating between near and far neighbors; this is known as *concentration effect.* This problem affects the Mahalanobis distance too, where the efforts in estimating the variance-covariance matrix should also be taken into account. For these reasons, with many variables the detection of multivariate outliers should be done focusing just on the subset of relevant continuous variables p' ($p' < p$).

5. APPLICATION OF THE CHOSEN METHODS TO SOME DATA FROM PANEL SURVEYS

This section investigates the performances of the methods presented in previous Sections when applied to a couple of datasets related to panel or pseudo-panel surveys that are described in Table 1.

Datasets used in the experiments

Table 1

Dataset/survey	Number of units	Type of units	Description
RDPerfComp	509	firms	R&D performing US manufacturing; yearly observations from 1982 to 1989 of the following variables: production, labor and capital[1]
RiceFarms	171	farms	Indonesian rice farm dataset, 171 farms producing rice observed 6 times. Many variables are available; the ones used in this study are: the total area cultivated with rice (in hectares); the total number of worked hours and the gross output of rice (in kg)[2].

In practice, in both the datasets the HB procedure is applied to each of the chosen variables in order to derive the corresponding scores ($E_{ci},\ i = 1,2,\dots,n;\ c = 1,\dots,p'$) provided by expression [2] with $U = 0.5$. These *E*-scores become the input of the following outlier detection techniques:

Md) robust Mahalanobis distance with adaptive distance threshold derived by comparing the theoretical distribution ($\chi^2_{p',0.975}$) with the empirical distribution of the squared robust distances (function `dd.plot()` in **mvoutlier** package);

SM) fit of a mixture of Gaussian distributions; outliers identified as "contaminated" units having a posterior probability greater than 0.5 (function `ml.est()` in **SeleMix** package with argument `tau=0.5`);

kNNw) *k*-NN weight (sum of *k*-NN distances; function `kNNdist()` in package **dbscan**) considering Euclidean distance and variables scaled in advance with a robust estimate of the standard deviation (based on inter-quartile range); approximate distance threshold set using the Kneedle algorithm;

LOF) Local Outlier Factor (`LOF()` function from **DDoutlier** package) with respectively $k = 5$ and $k = 10$; Euclidean distance is calculated on variables scaled in advance with a robust estimate of the standard deviation based on inter-quartile range; approximate distance threshold set using the Kneedle algorithm;

1. https://www.nuffield.ox.ac.uk/users/bond/index.html. See also the R package pder https://CRAN.R-project.org/package=pder

2. R package plm https://cran.r-project.org/package=plm

DBS) DBSCAN clustering algorithm with $g = 2p'$ and ε (distance threshold) set by using the Kneedle algorithm (function `dbscan()` in the package **dbscan**); Euclidean distance is calculated on the chosen variables scaled in advance with a robust estimate of the standard deviation based on inter-quartile range;

EIF) Extended isolation forest (with $o = 2$, no sub-sampling and 5000 trees in each forest; function `isolation.forest()` in the package **isotree**).

All the variables observed on the firms in the RDPerfComp dataset (production, labour and capital; $p' = p = 3$) are simultaneously considered; for each variable the *E*-scores are derived by comparing values observed in year 1986 vs. those in 1985.

Scores and outliers given by model-based, DBSCAN and Extend Isolation Forest outlier detection approaches with RDPerfComp dataset

Figure 1

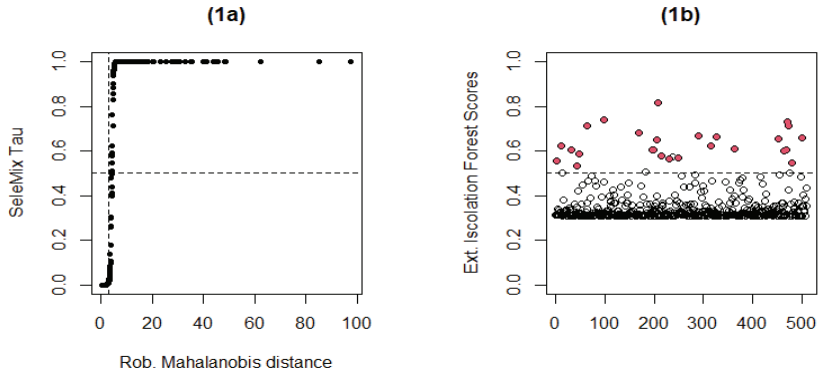


Figure 1a reports the scatterplot of the posterior probabilities estimated ("Tau") by SM vs. the robust Mahalanobis distance calculated by Md; the dashed vertical and horizontal lines indicate the corresponding thresholds and show that Md would return a higher number of outliers than SM.

Figure 1b shows the scores of EIF whereas the horizontal dashed line corresponds to the "standard" threshold $u_0 = 0.5$; the red-color filled dots indicate "noisy" observations (outliers) identified by DBS. Both the approaches return almost the same results, while the number of identified outliers is smaller if compared to that of SM or Md.

Table 2 compares result of approaches providing a direct identification of potential outliers (Md, SM, DBS) with the discretized scores of EIF. As already shown in Fig. 1b, DBS and EIF (when the rule-of-thumb $u_0 > 0.5$ is considered) point to almost the same relatively few outliers (27). The same units would be identified by Md and SM, that however, if jointly considered would return a nonnegligible number of additional potential outliers (97).

Outliers given by model-based, DBSCAN and Extend Isolation Forest outlier detection approaches with RDPerfComp dataset

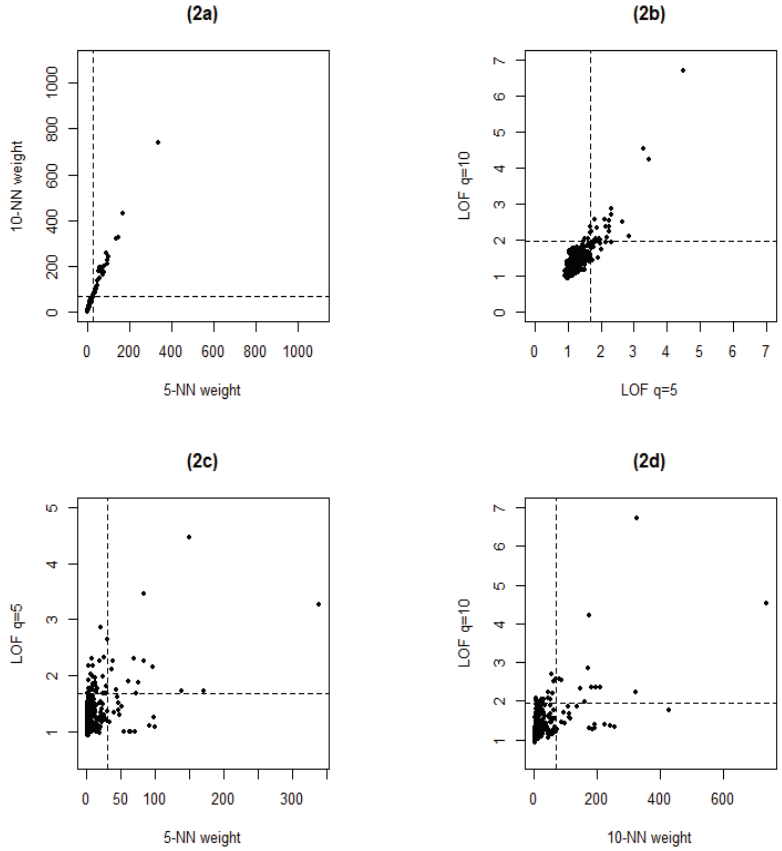
Table 2

			Scores EIF				
Md	SM	DBS	(0.3, 0.4]	(0.4, 0.5]	(0.5, 0.6]	(0.6, 0.7]	(0.7, 1.0]
Not-Outlier	Not-outlier	Not-outlier	351	0	0	0	h0
Outlier	Not-Outlier	Not-outlier	31	0	0	0	0
		Outlier	0	0	0	0	0
	Outlier	Not-outlier	58	39	3	0	0
		Outlier	0	0	8	14	5

Figure 2 summarizes results of kNNw and LOF. Figure 2a compares the scores obtained with $k = 5$ with those related to $k = 10$; the vertical and horizontal dashed lines denote the corresponding thresholds identified by applying the Kneedle algorithm; setting $k = 10$ gives a slightly higher number of potential outliers (5) than with $k = 5$; as in the expectations, the choice of *k* does not affect the results markedly. Similarly, Figure 2b compares the scores given by LOF with $q = 5$ and $q = 10$; the dashed lines indicate, as usual, the thresholds estimated with the Kneedle algorithm; the threshold identified with $q = 5$ returns a higher number of possible outliers compared to $q = 10$. Finally, Figures 2c and 2d compares scores of kNNw with those of LOF with respectively $k = q = 5$ and $k = q = 10$. In both the cases there is a high fraction of observations that are judged differently by the compared techniques.

Scores and outliers given by distance-based outlier detection approaches with RDPerfComp dataset

Figure 2



Figures 3 and 4 summarize the results obtained by applying outlier detection techniques to the panel of farms producing rice ("RiceFarms" dataset), when considering only three of the available continuous variables ($p' = 3 < 15 = p$): the total area cultivated with rice (in hectares); the total number of worked hours and the gross output of rice (in kg). In calculating the HB E-scores the 3rd and 4th observation occasions are considered.

Figure 3a shows the posterior probabilities estimated by SM vs. the robust Mahalanobis distance calculated by Md; as observed in the previous plots, the dashed lines indicate the corresponding thresholds. Also in this case study, Md returns a higher number of potential outliers if compared to SM.

Figure 3b shows that EIF scores are concentrated around the value 0.35 and there are relatively few potential outlying observations above the horizontal dashed line (corresponding to the $u_0 = 0.5$ threshold). In this case DBS with a distance threshold decided according to the Kneedle algorithm identifies just 5 "noisy" observations (outliers) (see also Table 3).

Scores and outliers given by model-based, DBSCAN and Extend Isolation Forest outlier detection approaches with RiceFarms dataset

Figure 3

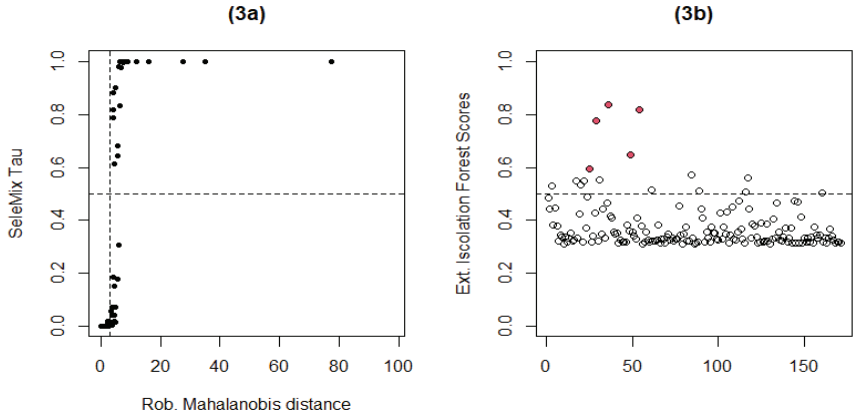


Table 3 counts the number of potential outliers obtained when crossing outcomes of Md, SM, DBS and the categorized scores of EIF. In this case study DBS identifies only 5 noisy observations out of 16 potential outliers given by EIF with the rule-of-thumb $u_0 > 0.5$. As in the previous example Md identifies a higher number of potential outliers if compared to the other procedures.

Outliers given by model-based, DBSCAN and Extend Isolation Forest outlier detection approaches with RiceFarm dataset

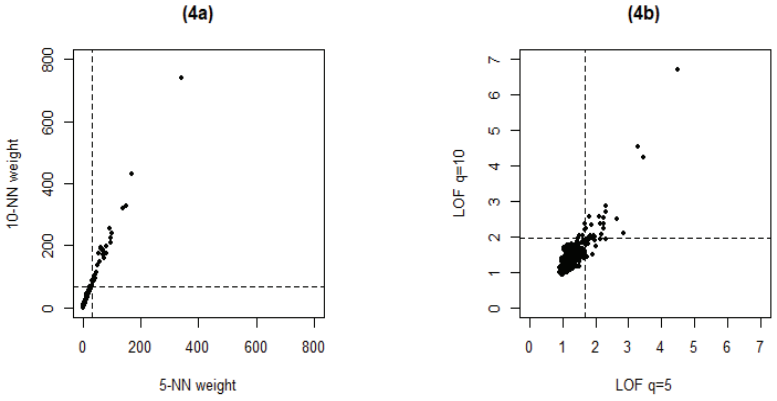
Table 3

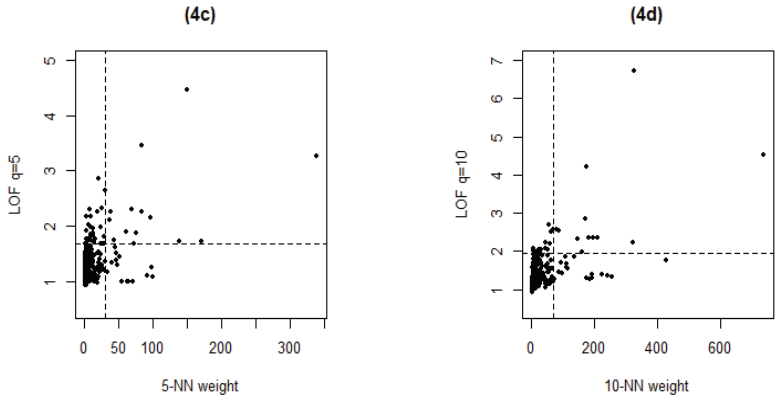
			Scores EIF				
Md	SM	DBS	(0.3, 0.4]	(0.4, 0.5]	(0.5, 0.6]	(0.6, 0.7]	(0.7, 1.0]
Not-Outlier	Not-outlier	Not-outlier	127	1	0	0	0
Outlier	Not-Outlier	Not-outlier	4	14	2	0	0
		Outlier	0	0	0	0	0
	Outlier	Not-outlier	0	9	9	0	0
		Outlier	0	0	1	1	3

Figure 4 summarizes outcomes of kNNw and LOF. Figure 4a compares the scores of kNNw obtained with the chosen values of k (5 and 10); the vertical and horizontal dashed lines denote the corresponding thresholds identified by applying the Kneedle algorithm. Also in this case, increasing the value of k gives a slightly higher number of potential outliers. Figure 4b compares the scores given by LOF with $q = 5$ and $q = 10$, again the lower value of q returns a higher number of possible outliers compared to $q = 10$. Finally, Figures 2c and 2d compares scores of kNNw with those of LOF with respectively $k = q = 5$ and $k = q = 10$. Both the scatterplots show that a high fraction of observations is judged differently by the compared techniques, as in the case of firms' dataset.

Scores and outliers given by distance-based outlier detection approaches with RiceFarms dataset

Figure 3





6. CONCLUSIONS

This short note compares a series of approaches to detect multivariate outliers in particular when dealing with panel survey data and the objective consists also in measuring change over time. In particular, we search for multivariate outliers by analyzing the *E*-scores derived from the initial ratios of values observed in subsequent time occasions ($r_i = y_{t_2 i}/y_{t_1 i}$), as proposed by Hidiroglou-Berthelot; this way of working has the advantage of reducing the dimensionality of the data and tailor the outlier detection towards ratios involving large quantities that may have a higher influence on the final survey estimates.

The model-based outlier detection approaches considering a multivariate Gaussian distribution – robust Mahalanobis distance from the bulk of the data (Md) and fitting of mixture of Gaussian distributions (SM) – permit a direct identification of the potential outliers (units above thresholds) but in our small empirical study they tend to identify a larger number of potential outliers than the other approaches; in any case in both the approaches it is possible to use the resulting "score" (respectively robust Mahalanobis distance and posterior estimated probabilities of being outliers) to sort observations in decreasing order and start inspecting them with the support of subject matter experts. Both the approaches do not need to scale the variables in advance; they are quite easy to apply although fitting a mixture of distributions require setting a number of parameters related to the estimation process.

The nonparametric approaches based on calculating the distances are quite straightforward to apply; our study uses the Euclidean distance function (default in all the considered R functions) calculated on the *E*-scores, scaled in advance by a robust estimate of the standard deviation to account for different variability. In all the cases we decided to estimate approximately the threshold

needed to identify potential outliers by applying the Kneedle algorithm, as sorted distance-based scores usually show an increasing concave-up curve. The algorithm is quite simple and easy to be coded in R.

In both the considered datasets, increasing the value of k in the k-NN weight, coupled with the chosen strategy for setting the threshold, tends to provide a higher number of potential outliers. Outcomes of LOF seem in contrast with those of k-NN weight; this seems due to the fact that the chosen datasets do not include sub-groups of observations with different local densities.

The DBSCAN clustering method uses also the Euclidean distances calculated on scaled E-scores. The key role is played by ε – the distance threshold – that in our empirical study is decided by applying the Kneedle algorithm; this latter decision led to identify relatively few units with a high potential of being outliers in both the considered datasets. The results of BDSCAN are quite aligned with those of the Extended Isolation Forest (that in our application partitions units according to randomly generated regression lines). The EIF approach has the advantage of avoiding to transform the variables in advance and has relatively few tuning parameters to set that, however, can be decided without resorting to additional algorithms/methods; in addition, the final scores range between 0 and 1 simplifying their analysis. It is worth noting that the rule-of-thumb consisting in identifying as potential outliers those units whose score is grater that 0.5 should be applied carefully; this limited empirical study seems to suggest that a slight inferior value, between 0.4 and 0.5 may work better; this finding need to be confirmed by additional investigation.

In summary, the limited empirical comparison carried out in this work, jointly to the results obtained by D'Orazio (2023) in the univariate case, show that the Isolation Forest and its extended version in the multivariate case (EIF) represent a promising approach for detecting potential multivariate outliers in official statistics with the great advantage that the results do not seem markedly affected the starting tuning parameters even if the rule-of-thumb of considering as potential outliers observation with a score greater than 0.5 should be applied carefully.

More in general, this study further confirms, if still needed, that the R environment is a fundamental software package for official statistics as it offers implementations of both "traditional" and recent statistical methods, as shown in this study limited to outlier detection procedures.

References

1. **Alfons A, Templ M** (2013). *"Estimation of Social Exclusion Indicators from Complex Surveys: The R Package laeken."* Journal of Statistical Software, 54, pp. 1–25.
2. **Andreas Alfons, A., Templ, M., Filzmoser, P.** (2012) *"Robust estimation of economic indicators from survey samples based on Pareto tail modelling*", Journal of the Royal Statistical Society. Series C (Applied Statistics), 62, pp. 271-286
3. **Angiulli, F. Pizzuti, C**., (2002) "*Fast Outlier Detection in High Dimensional Spaces"*. Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery. Springer-Verlag, pp. 15–26.
4. **Breunig, M.M., Kriegel, H.-P., Ng, R. T., Sander, J.** (2000). *"LOF: Identifying Density-Based Local Outliers"*. Proceedings of the International Conference On Management of Data. Dallas, TX. pp. 93-104
5. **Campos, G. O. Zimek, A., Sander, J., Campello, R.J.G.B., Micenková, B., Schubert, E., Assent, I., Houle, M. E.** (2016) *"On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study",* Data Mining and Knowledge Discovery, 30, pp. 891–927.
6. **Cortes, D.** (2022) *isotree: Isolation-Based Outlier Detection*. R package version 0.5.15, https://CRAN.R-project.org/package=isotree
7. **D'Orazio, M.** (2022). *univOutl: Detection of Univariate Outliers*. R package version 0.3. https://CRAN.R-project.org/package=univOutl
8. **D'Orazio, M.** (2023) *"An empirical comparison of some outlier detection methods with longitudinal data"*, To be published in Istat's Working Paper series.
9. **Di Zio, M., Guarnera, U.** (2013) "A Contamination Model for Selective Editing", Journal of Official Statistics, 29, pp. 539-555.
10. **Ester, M., Kriegel, H.-P., Sander, J., Xu, X.** (1996) *"A density-based algorithm for discovering clusters in large spatial databases with noise"*. Proceedings of the 2nd Int. Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press, pp. 226–231
11. **Eurostat**, 2014. *Memobust Handbook on Methodology of Modern Business Statistics*. Luxembourg https://ec.europa.eu/eurostat/cros/content/handbook-methodology-modern-business-statistics_en
12. **Filzmoser, P., Gschwandtner, M.** (2021) *mvoutlier: Multivariate Outlier Detection Based on Robust Methods*. R package version 2.1.1, https://CRAN.R-project.org/package=mvoutlier
13. **Filzmoser, P., Garrett, R.G., Reimann, C.** (2005) *"Multivariate outlier detection in exploration geochemistry"*, Computers & Geosciences, 31, pp. 579–587.
14. **Guarnera, U., Buglielli, T**. (2020) *SeleMix: Selective Editing via Mixture Models*. R package version 1.0.2, https://CRAN.R-project.org/package=SeleMix.
15. **Hautamäki, V., Kärkkäinen, I., Fränti, P**. (2004) "*Outlier Detection Using k-Nearest Neighbour Graph*". International Conference on Pattern Recognition, pp. 430-433
16. **Hahsler, M., Piekenbrock, M.** (2022) d*bscan: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and Related Algorithms*. R package version 1.1-10, https://CRAN.R-project.org/package=dbscan
17. **Hahsler, M., Piekenbrock, M., Doran, D.** (2019). *"dbscan: Fast Density-Based Clustering with R"*. Journal of Statistical Software, 91, pp. 1-30
18. **Hariri, S., Kind, M. C., Brunner, R. J.** (2018) "*Extended Isolation Forest"* arXiv preprint arXiv:1811.02141.
19. **Hidiroglou, M.A., Berthelot, J.-M**. (1986) "*Statistical editing and Imputation for Periodic Business Surveys"*, Survey Methodology, 12, pp. 73-83.
20. **Hidiroglou, M.A. Emond, N.** (2018) *"Modifying the Hidiroglou-Berthelot (HB) method"*. Unpublished note, Business Survey Methods Division, Statistics Canada, May 18, 2018.

21. **Hubert, M., Van der Veeken, S.** (2008) *"Outlier Detection for Skewed Data"*. Journal of Chemometrics, 22, pp. 235-246.
22. **Hubert, M., Vandervieren, E**. (2008) "*An Adjusted Boxplot for Skewed Distributions*" Computational Statistics & Data Analysis, 52, pp. 5186-5201
23. **Liu, F.T., Ting, K.M., Zhou, Z**. (2008) *"Isolation forest".* In: Eighth IEEE International Conference on Data Mining, pp. 413–422
24. **Liu, F. T., Ting, K. M., Zhou, Z.** (2010) *"On detecting clustered anomalies using SCiForest"* Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, Berlin, Heidelberg, 2010.
25. **Liu, F. T., Ting, K. M., Zhou, Z.** (2012) *"Isolation-based anomaly detection"*. ACM Transactions on Knowledge Discovery from Data (TKDD), 6, pp. 1–39
26. **Madsen, J.H.** (2018) *DDoutlier: Distance & Density-Based Outlier Detection*. R package version 0.1.0. https://CRAN.R-project.org/package=DDoutlier
27. **Maechler. M., Rousseeuw, P., Croux, C., Todorov, V., Ruckstuhl, A., Salibian-Barrera, M., Verbeke, T., Koller, M., Conceicao, E.L.T., di Palma, M.A**. (2022). *robustbase: Basic Robust Statistics* R package version 0.95-0. http://CRAN.R-project.org/package=robustbase
28. **Maronna, R.A., Zamar, R.H**. (2002) *"Robust estimates of location and dispersion of high-dimensional datasets"*, Technometrics, 44, pp. 307–317.
29. **R Core Team** (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/
30. **Ramaswamy S., Rastogi, R., Shim, K.** (2000) *"Efficient Algorithms for Mining Outliers from Large Data Sets"*. Proceedings of the International Conference on Management of Data (SIGMOD '00), pp. 427-438.
31. **Rousseeuw, P.J., Croux, C**. (1993) *"Alternatives to the Median Absolute Deviation"*, Journal of the American Statistical Association, 88, pp. 1273–1283.
32. **Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B**. (2010) *"Finding a 'Kneedle' in a Haystack: Detecting Knee Points in System Behavior"*, Proceedings of the 30th International Conference on Distributed Computing Systems SIMPLEX Workshop (ICDCS 2010), Genoa, Italy http://www.icsi.berkeley.edu/pubs/networking/findingakneedle10.pdf
33. **Schubert, E., Sander, J., Ester, M., Kriegel, H.P., Xu, X.** (2017) *"DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN"*. ACM Trans. Database Syst., 42, pp. 19:1-19:21.
34. **Srikanth, K.** (2021). *solitude: An Implementation of Isolation Forest*. R package version 1.1.3, https://CRAN.R-project.org/package=solitude
35. **Todorov, V., Filzmoser, P.** (2009) *"An Object-Oriented Framework for Robust Multivariate Analysis"*. Journal of Statistical Software, 32, pp. 1–47.
36. **UNECE** (2000) *Glossary of Terms on Statistical Data Editing*. Geneva https://ec.europa.eu/eurostat/ramon/statmanuals/files/UN_editing_glossary_2000.pdf
37. **Waal, T-de, Pannekoek, J., Scholtus, S**. (2011) *Handbook of statistical data editing and imputation*. John Wiley & Sons, Inc., Hoboken.
38. **van der Loo, M.P.J**., (2010) *"Distribution based outlier detection for univariate data"*, Discussion paper 10003, Statistics Netherlands, The Hague.
39. **Zimek, A. Filzmoser, P.** (2018) *"There and back again: Outlier detection between statistical reasoning and data mining algorithms"*. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8, e1280.
40. **Zimek, A., Schubert, E., Kriegel, H.-P.** (2012) *"A survey on unsupervised outlier detection in high-dimensional numerical data"*. Statistical Analysis and Data Mining, 5, pp. 363–387.