Behavior Research Methods
https://doi.org/10.3758/s13428-024-02356-w

ORIGINAL MANUSCRIPT

# Check your outliers! An introduction to identifying statistical outliers in R with *easystats*

**Rémi Thériault[1] · Mattan S. Ben-Shachar[2] · Indrajeet Patil[3] · Daniel Lüdecke[4] · Brenton M. Wiernik[5] · Dominique Makowski[6]**

Accepted: 2 February 2024


**Abstract**
Beyond the challenge of keeping up to date with current best practices regarding the diagnosis and treatment of outliers, an additional difficulty arises concerning the mathematical implementation of the recommended methods. Here, we provide an overview of current recommendations and best practices and demonstrate how they can easily and conveniently be implemented in the R statistical computing software, using the *{performance}* package of the *easystats* ecosystem. We cover univariate, multivariate, and model-based statistical outlier detection methods, their recommended threshold, standard output, and plotting methods. We conclude by reviewing the different theoretical types of outliers, whether to exclude or winsorize them, and the importance of transparency. A preprint of this paper is available at: 10.31234/osf.io/bu6nt.

**Keywords** Univariate outliers · Multivariate outliers · Robust detection methods · R · Easystats

## Introduction

Real-life data often contain observations that can be considered *abnormal* when compared to the main population. The cause of this *abnormality* can be hard to assess and the boundaries of "normal" difficult to define—they may truly belong to a different distribution (originating from a different generative process) or simply be extreme cases, statistically rare but not impossible.

Nonetheless, the improper handling of these outliers can substantially affect estimation of quantities of interest, and, in the context of statistical models, can bias parameter estimates and weaken a model's predictive performance (Aguinis et al., 2013). It is thus essential to address this problem thoughtfully. Yet, despite the existence of established recommendations and guidelines, many researchers still do not treat outliers consistently, or do so using inappropriate strategies (Aguinis et al., 2013; Leys et al., 2013; Simmons et al., 2011).

Understanding the various methods for outlier detection, their differences, as well as their benefits and disadvantages, can aid researchers in choosing between them and applying them correctly (see Smiti, 2020, for an overview of pros and cons of several recently developed advanced methods). For example, Fig. 1 shows a hypothetical dataset of women's heights and weights (based on the "women" dataset in R; McNeil, 1977) and how applying three different types of outlier identification methods (univariate, multivariate, and model-based; all described in detail in this paper) can lead to different results.

One possible reason researchers do not employ validated strategies is that they may not be aware of existing recommendations, or do not know how to implement them using their analysis software. In this paper, we show how to follow current best practices for automatic and reproducible statistical outlier detection (SOD) using R and the *{performance}* package (Lüdecke et al., 2021), which is part of the *easystats* ecosystem of packages that build an R framework

✉ Rémi Thériault
theriault.remi@courrier.uqam.ca

1 Department of Psychology, Université du Québec à Montréal, Succursale Centre-Ville, C.P. 8888, Montréal, Québec H3C 3P8, Canada

2 Independent Researcher, Ramat Gan, Israel

3 Center for Humans and Machines, Max Planck Institute for Human Development, Berlin, Germany

4 Institute of Medical Sociology, University Medical Center Hamburg-Eppendorf, Hamburg, Germany

5 Independent Researcher, Tampa, FL, USA

6 School of Psychology, University of Sussex, Brighton, UK

Published online: 25 March 2024

for easy statistical modeling, visualization, and reporting (Lüdecke et al., 2023). Installation instructions can be found on GitHub or its website, and its list of dependencies on CRAN.

The instructional materials that follow are aimed at an audience of researchers who want to follow good practices, and are appropriate for advanced undergraduate students, graduate students, professors, or professionals having to deal with the nuances of outlier treatment.

## Identifying outliers

Although many researchers attempt to identify outliers with measures based on the mean (e.g., $z$ scores), those methods can be problematic. This is because the mean and standard deviation themselves are not robust to the influence of outliers and those methods also assume normally distributed data (i.e., a Gaussian distribution). Therefore, current guidelines recommend using robust methods to identify outliers, such as those relying on the median as opposed to the mean (Leys et al., 2013, 2018, 2019). Additionally, univariate methods can give false positives since they ignore the patterns in multidimensional data, which are often of interest (such as comparing conditional means or estimating correlation matrices). In such cases, multivariate outlier detection methods may be of relevance.

Which exact outlier method to use depends on many factors. In some cases, eye-gauging odd observations can be an appropriate solution, though many researchers will favor algorithmic solutions to detect potential outliers, for example, based on a continuous value expressing the observations that stand out from the others. Indeed, relying on human intuition and "visual checks" can be rather subjective, and sometimes, even suboptimal. For example, visually communicating results containing outliers—say, on a scatter plot—has been shown to bias people's estimations of a regression line, even when individuals correctly detect the outliers (Ciccione et al. 2023).

One of the factors to consider when selecting an algorithmic outlier detection method is the statistical test of interest. Identifying observations where the regression model does not fit well can help find information relevant to our specific research context. This approach, known as model-based outlier detection (as outliers are extracted after the statistical model has been fit), can be contrasted with distribution-based outlier detection, which is based on the distance between an observation and the "center" of its population. Various quantification strategies of this distance exist for the latter, both univariate (involving only one variable at a time) and multivariate (involving multiple variables).

However, we would like to emphasize that the methods listed in this paper are not an exhaustive list of methods developed and available to researchers. For instance, Bayesian approaches that do not fully reject outliers but simply lower their "weights" have been partly formalized by Chaloner and Brant (1988) and recently implemented by Ciccione et al. (2023). Crucially, Ciccione and colleagues also provide empirical evidence that human observers might indeed perform such forms of Bayesian re-weighting of outliers when asked to detect and reject them, making interesting parallels between statistical research methods and naive psychological mechanisms.

Importantly, whatever approach researchers choose remains a subjective decision, and usage (and rationale) must be transparently documented and reproducible (Leys et al., 2019). Researchers should commit (ideally in a preregistration) to an outlier treatment method before collecting the data. They should report in the paper their decisions and details of their methods, as well as any deviation from their original plan. These transparency practices can help reduce false positives due to excessive researchers' degrees of freedom (i.e., choice flexibility throughout the analysis). In the following section, we go through each of the mentioned methods and provide examples of how to implement them with R.

## Univariate outliers

Researchers frequently attempt to identify outliers using measures of deviation from the center of a variable's distribution. One of the most popular of such procedures is the $z$-score transformation, which computes the distance in standard deviation (SD) from the mean. However, as mentioned earlier, this method is not robust. Therefore, for univariate outliers, it is recommended to use the median along with the median absolute deviation (MAD), which is more robust than the interquartile range or the mean and its standard deviation (Leys et al., 2013, 2019).

Researchers can identify outliers based on robust (i.e., MAD-based) $z$ scores using the `check_outliers()` function of the *{performance}* package, by specifying `method = "zscore_robust"`.[1] Although Leys et al. (2013) suggest a default threshold of 2.5 and Leys et al. (2019) a threshold of 3, *{performance}* uses by default a less conservative threshold of $\approx 3.29$.[2] That is, data points will be flagged as outliers if they go beyond $\pm \approx 3.29$ MAD. Users can adjust this threshold using the threshold argument.

Below, we provide example code using the `mtcars` dataset, which was extracted from the 1974 *Motor Trend*

[1] Note that `check_outliers()` only checks numeric variables.

[2] 3.29 is an approximation of the two-tailed critical value for $p < .001$, obtained through `qnorm(p=1-0.001/2)`. We chose this threshold for consistency with the thresholds of all our other methods.

US magazine. The dataset contains fuel consumption and ten characteristics of automobile design and performance for 32 different car models (see `?mtcars` for details). We chose this dataset because it is accessible from base R and familiar to many R users. We might want to conduct specific statistical analyses on this data set, say, *t* tests or structural equation modeling, but first, we want to check for outliers that may influence those test results.

Because the automobile names are stored as column names in `mtcars`, we first have to convert them to an ID column to benefit from the `check_outliers()` ID argument. Furthermore, we only really need a few columns for this demonstration, so we pick the first four (mpg = Miles/(US) gallon; cyl = Number of cylinders; disp = Displacement; hp = Gross horsepower). Finally, because there are no outliers in this dataset, we add two artificial outliers before running our function.

```
library(performance)

# Create some artificial outliers and an ID column
data <- rbind(mtcars[1:4], 12, 55)
data <- cbind(car = row.names(data), data)
outliers <- check_outliers(data, method = "zscore_robust", ID = "car")
outliers

#> 1 outlier detected: case 34.

#> - Based on the following method and threshold: zscore_robust (3.291).
#> - For variables: mpg, cyl, disp, hp.
#>
#> ----------------------------------------------------------------------------

#>
#> The following observations were considered outliers for two or more
#>   variables by at least one of the selected methods:
#>
#>   Row car n_Zscore_robust
#> 1  34  34               2
#>
#> ----------------------------------------------------------------------------

#> Outliers per variable (zscore_robust):
#>
#> $mpg
#>    Row car Distance_Zscore_robust
#> 34  34  34               6.271888
#>
#> $cyl
#>    Row car Distance_Zscore_robust
#> 34  34  34               16.52502
#>
```

What we see is that `check_outliers()` with the robust *z* score method detected one outlier: cases 34, which is one of the observations we added ourselves. It was flagged for two variables specifically: mpg (miles/(US) gallon) and cyl (number of cylinders), and the output provides its exact *z*-score for those variables.

We describe how to deal with outliers in more details later in the paper, but should we want to exclude detected outliers from the main dataset, we can extract row numbers using `which()` on the output object, which can then be used for indexing:

```
which(outliers)

#> [1] 34

data_clean <- data[-which(outliers), ]
```

All `check_outliers()` output objects possess a `plot()` method, meaning it is also possible to visualize all observations in a way that highlights the outliers using the generic `plot()` function on the resulting outlier object after loading the {*see*} package (Fig. 2).

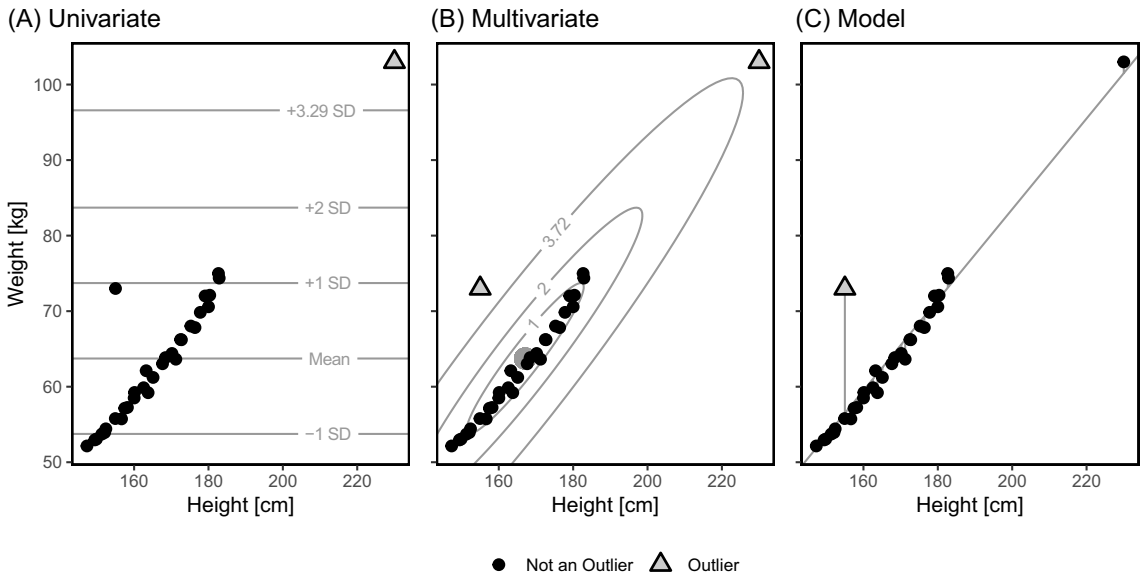


**Fig. 1** Visual representation for the most common methods in the families of outlier identification applied to a hypothetical dataset of women's heights and weights. *Note.* In each subplot, *triangles* are observations marked as "outliers". **A** Univariate method: Observations are marked as outliers if they lie at some fixed or relative distance from the center of each variable (in this case, 3.29 standard deviations from *y*'s mean), suggesting they are not part of the same distribution as the rest of the data; **B** Multivariate method: Observations are marked as outliers if they lie at some fixed or relative distance from the multivariate center (in this case, a Mahalanobis distance of 3.72 from the centroid defined by the means of *x* and *y*), suggesting they are not part of the same multivariate distribution as the rest of the data; **C** Model-based method: Observations are marked as outliers if they affect the model's estimated parameters by more than some threshold (in this case, they have a Cook's distance of 0.71), suggesting that the inclusion of such observations biases the estimated parameters to a large degree (in the plot, this is represented as the observation with the large absolute residual [i.e., the distance from the regression line]—a concept closely related to Cook's distance). As can be seen, although there is some overlap, the three methods do not agree on which observations are to be marked as outliers. Code to reproduce this figure and all analyses is available at https://osf.io/eqja6/

```
library(see)

plot(outliers)
```

Other univariate methods are available, such as using the interquartile range (IQR), or based on different intervals, such as the highest density interval (HDI) or the bias-corrected and accelerated interval (BCI). These methods are documented and described in the function's help page.

## Multivariate outliers

Univariate outliers can be useful when the focus is on a particular variable, for instance the reaction time, as extreme values might be indicative of inattention or non-task-related behavior.[3]

However, in many scenarios, the variables of a data set are not independent, and an outlying observation or participant will be reflected to various degrees on multiple variables. For instance, in the case of survey studies containing a large number of items (e.g., many Likert scales), "careless" or low-effort responding participations (e.g., participants answering at random, displaying "straight-lining", or "zigzagging" patterns of response) becomes more common—especially when relying on online samples such as through MTurk (Aruguete et al., 2019; Goldammer et al., 2020; Ward & Meade, 2023). Although specific methods exist to detect these unwanted behaviors in questionnaires (e.g., Cao et al., 2018; Curran, 2016; Yentes & Wilhelm, 2023; Zijlstra et al., 2011), this issue can be framed more generally as follows: multiple "odd" observations can sum up and reveal an abnormal participant. Importantly, the deviation from the norm could potentially be low for all variables when taken independently (not meeting the rejection criteria), but strong when taken together (in other words, the likelihood of being an outlier on one variable can be independent from the probability of being an outlier on multiple variables).

One common approach for this is to compute multivariate distance metrics, such as the Mahalanobis distance.

[3] Note that univariate outlier detection methods might not be the optimal way of treating reaction time outliers (Ratcliff, 1993; Van Zandt & Ratcliff, 1995).

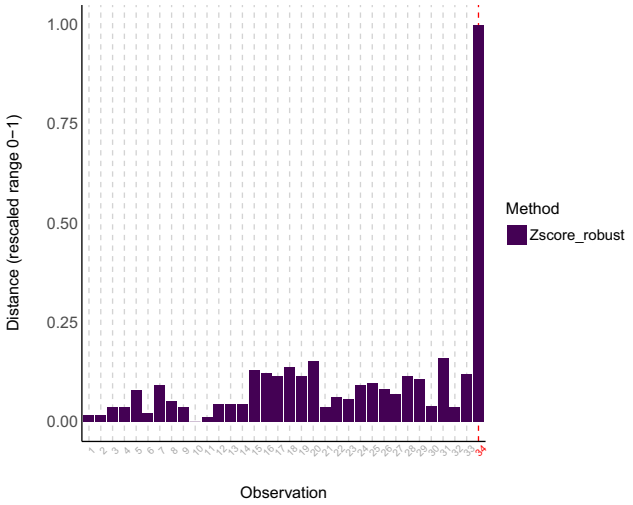


**Fig. 2** Visual depiction of outliers using the robust *z*-score method. *Note.* The distance represents the highest deviation score per participant for variables mpg, cyl, disp, and hp. This score represents a given participant's (1–34) highest robust *z* score among the tested variables. The resulting unique value (representing one of mpg, cyl, disp, or hp for that participant) is then rescaled to a range of 0 to 1 by dividing by the value of the participant with the highest score

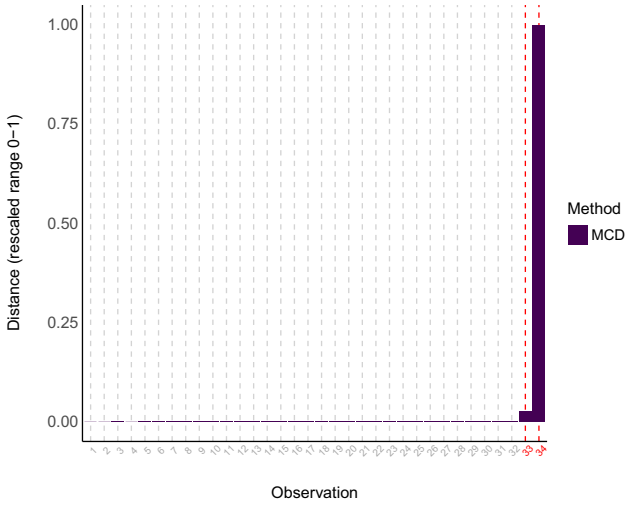


**Fig. 3** Visual depiction of outliers using the minimum covariance determinant (MCD) method. *Note.* The minimum covariance determinant (MCD) method is a robust version of the Mahalanobis distance. The distance represents the MCD scores for variables mpg, cyl, disp, and hp

Although the Mahalanobis distance is very popular, just like the regular *z* scores method, it is not robust and is heavily influenced by the outliers themselves. Therefore, for multivariate outliers, it is recommended to use the minimum covariance determinant, a robust version of the Mahalanobis distance (MCD, Leys et al., 2018, 2019).

In *{performance}*'s `check_outliers()`, one can use this approach with `method = "mcd"`.[4]

```
outliers <- check_outliers(data, method = "mcd")
outliers

#> 2 outliers detected: cases 33, 34.
#> - Based on the following method and threshold: mcd (20).
#> - For variables: mpg, cyl, disp, hp.
```

Here, we detected nine multivariate outliers (i.e., when looking at all variables of our dataset together). We can see the result in Fig. 3.

```
model <- lm(mpg ~ disp * hp, data = data)
outliers <- check_outliers(model, method = "cook")
outliers
```

In small samples, however, the MCD method tends to be inaccurate, especially when dealing with high-dimensional data. Other multivariate methods are also available, such as the classic Mahalanobis distance and another type of robust Mahalanobis distance that relies on an orthogonalized Gnanadesikan–Kettenring pairwise estimator (Gnanadesikan & Kettenring, 1972). These methods are documented and described in the function's help page.

## Model-based outliers

Working with regression models creates the possibility of using model-based SOD methods. These methods rely on the concept of *leverage*, that is, how much influence a given observation can have on the model estimates. If few observations have a relatively strong leverage/influence on the model, one can suspect that the model's estimates are biased by these

[4] Our default threshold for the MCD method is defined by `stats::qchisq(p=1-0.001, df=ncol(x))`, which again is an approximation of the critical value for $p < .001$ consistent with the thresholds of our other methods.

observations, in which case flagging them as outliers could prove helpful (see next section, "Handling outliers").

In {performance}, two such model-based SOD methods are currently available: Cook's distance, for regular regression models, and Pareto, for Bayesian models. As such, `check_outliers()` can be applied directly on regression model objects, by simply specifying `method="cook"` (or `method="pareto"` for Bayesian models).[5]

Currently, most `lm` models are supported (except for `glmmTMB`, `lmrob`, and `glmrob` models), as long as they are supported by the underlying functions `stats::cooks.distance()` (or `loo::pareto_k_values()`) and `insight::get_data()` (for a full list of the 225 models currently supported by the {*insight*} package, see https://easystats.github.io/insight/#list-of-supported-models-by-class). We show a demo below.

```
#> 1 outlier detected: case 33.
#> - Based on the following method and threshold: cook (0.806).
#> - For variable: (Whole model).
```

Using the model-based outlier detection method, we identified a single outlier. We can see the result in Fig. 4.

```
plot(outliers)
```

Table 1 below summarizes which methods to use in which cases, and with what threshold. The recommended thresholds are the default thresholds.

Leys et al. (2018) report a preference for the MCD method over Cook's distance. This is because Cook's distance removes one observation at a time and checks its corresponding influence on the model each time (Cook, 1977), and flags any observation that has a large influence. In the view of these authors, when there are several outliers, the process of removing a single outlier at a time is problematic as the model remains "contaminated" or influenced by other possible outliers in the model, rendering this method suboptimal in the presence of multiple outliers.

However, distribution-based approaches are not a silver bullet either, and there are cases where the usage of methods agnostic to theoretical and statistical models of interest might be problematic. For example, a very tall person would be expected to also be much heavier than average, but that would still fit with the expected association between height and weight (i.e., it would be in line with a model such as weight ~ height). In contrast, using multivariate outlier detection methods in such a case may flag this person as being an outlier—being unusual on two variables, height and weight—even though the pattern fits perfectly with our predictions.

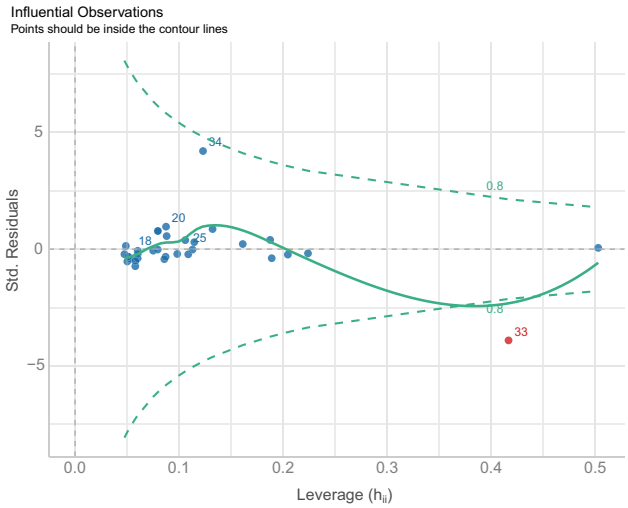


**Fig. 4** Visual depiction of outliers based on Cook's distance (leverage and standardized residuals). *Note.* This plot is based on the fitted model

Refer again to Fig. 1: In panel B, both an extremely tall woman, and a shorter but heavier woman are flagged as outlier due to their (Mahalanobis) distance from the group's centroid. However, when examined in the context of the relationship between height and weight (panel C), it is clear that the taller woman's weight falls along the regression line. That is, it is *model-consistent*—we expect an extremely tall person to weigh more, and so this observation is not marked as an outlier using a model based method, though it is when using univariate (panel A) or multivariate (panel B) methods. On the other hand, the second observation not only has a high Cook's distance, meaning it has influenced the model's estimates by a large degree, but it also clearly diverges from the regression line—it is *model-inconsistent*, and is accordingly flagged as an outlier.

This model-based approach to outlier detection is most coherent in regression-based settings; however, sometimes we *are*

[5] Our default threshold for the Cook method is defined by `stats::qf(0.5, ncol(x), nrow(x) - ncol(x))`, which again is an approximation of the critical value for $p < .001$ consistent with the thresholds of our other methods. In this case, the value 0.5 represents the median of the implied $F$ distribution for $D$, which allows us to flag $D$ values that are "above average".

**Table 1** Summary of statistical outlier detection methods recommendations

| Statistical test | Diagnosis method | Recommended threshold | Function usage |
|---|---|---|---|
| Supported regression model | Model-based: Cook (or Pareto for Bayesian models) | `qf(0.5, ncol(x), nrow(x) - ncol(x))` (or 0.7 for Pareto) | `check_outliers(model, method="cook")` |
| Structural Equation Modeling (or other unsupported model)[1] | Multivariate: Minimum Covariance Determinant (MCD) | `qchisq(p=1-0.001, df=ncol(x))` | `check_outliers(data, method="mcd")` |
| Simple test with few variables (*t* test, correlation, etc.) | Univariate: robust z scores (MAD) | `qnorm(p=1-0.001/2)`, ≈ 3.29 | `check_outliers(data,method="zscore_robust")` |

[1]The Minimum Covariance Determinant (MCD) can be inaccurate for small sample sizes. In these cases, the classic Mahalanobis distance can be used instead

interested in multi-dimensional outlier detection in the classical sense of a point that is far away from the general cluster of our data. We might, for example, decide to exclude a person who is extremely tall and heavy because they differ too much from the main population of study, even if they *do* match the general trend. In these cases, other methods such as MCD can be appropriate.

Finally, unusual observations happen naturally: extreme observations are expected even when taken from a normal distribution. While statistical models can integrate this "expectation", multivariate outlier methods might be too conservative, flagging too many observations despite belonging to the right generative process. For these reasons, we believe that model-based methods are still preferable to the MCD when using supported regression models. Additionally, if the presence of multiple outliers is a significant concern, regression methods that are more robust to outliers should be considered—like *t* regression or quantile regression—as they render their precise identification less critical (McElreath, 2020).

## Composite outlier score

To reiterate, there is not any wrong method, per se. Different methods can be judged by their usefulness to do *something*, but do so differently. Univariate methods are often good at detecting non-representative values or data-coding errors. Multivariate methods are also good at detecting non-representative values in a joint-distribution sense. Similarly, model-based methods are good for detecting values that might unrealistically bias model inference.

The *{performance}* package offers a consensus-based approach that combines several methods, based on the assumption that different methods provide different angles of looking at a given problem. By applying a variety of methods, one can hope to "triangulate" the "true" outliers (those consistently flagged by multiple methods) and thus attempt to minimize false positives.

In practice, this approach computes a composite outlier score, formed of the average of the binary (0 or 1) classification results of each method. It represents the probability that each observation is classified as an outlier by at least one method. The default decision rule classifies rows with composite outlier scores superior or equal to 0.5 as outlier observations (i.e., that were classified as outliers by at least half of the methods). In *{performance}*'s `check_outliers()`, one can use this approach by including all desired methods in the corresponding argument. Returning to the example model above:

```
outliers <- check_outliers(model, method = c("zscore_robust", "mcd", "cook"))
which(outliers)

#> [1] 33 34
```

Outliers (counts or per variables) for individual methods can then be obtained through attributes. For example:

```
attributes(outliers)$outlier_var$zscore_robust

#> $mpg
#>    Row Distance_Zscore_robust
#> 34  34               6.271888
#>
```

An example sentence for reporting the usage of the composite method could be:

> Based on a composite outlier score (see the '`check_outliers()`' function in the *{performance}* R package, Lüdecke et al., 2021) obtained via the joint application of multiple outliers detection algorithms ((a) median absolute deviation (MAD)-based robust $z$ scores, Leys et al., 2013; (b) Mahalanobis minimum

covariance determinant (MCD), Leys et al., 2019; and (c) Cook's distance, Cook, 1977), we excluded two participants that were classified as outliers by at least half of the methods used.

## Handling outliers

The above sections demonstrated how to identify outliers using the `check_outliers()` function in the *{performance}* package. But what should we do with these outliers once identified? It is common to automatically discard any observation that has been marked as "an outlier" as if it might infect the rest of the data with its statistical ailment. However, it is important to remember that researchers do not have access to the ground truth—it is not possible to know which observations truly do not "belong" with the rest of the sample. Instead, outlier detection methods behave much like unsupervised learning methods, trying to find patterns in the data, and to mark observations that seem to have a bad "fit" with these patterns.

Therefore, we believe that these methods should merely be used as suggestive, and advocate for researchers and analysts to use their *domain knowledge* when deciding how to deal with observations marked as outliers using SOD. Indeed, automatic tools can help detect outliers, but they are nowhere near perfect. Although they can be useful for flagging suspect data, they can have misses and false alarms, and they cannot completely replace human eyes and proper vigilance from the researcher. That is, the use of SOD methods is but one step in the get-to-know-your-data pipeline.

For example, in the case of reaction time analysis, Miller (2023) systematically compared 58 SOD procedures in simulations using large datasets of real reaction times. He concluded that regardless of the selected procedure, the exclusion of outliers (reaction times too slow or too fast) generally did more harm than good compared to retaining them, as they tend to incorrectly detect outliers, reduce statistical power, and increase bias and noise. He thus recommends only excluding invalid reaction times, such as those under a fixed threshold, e.g., 150 ms, which is close to the minimal physiological limit for reacting to a visual stimulus. Setting an upper limit on very long times (e.g. , 3–5 s, depending on the experimental task) to remove potential sparse artifacts can also improve model convergence and fitting.

Miller (2023) also suggests that it is typically better to assess outliers within specific experimental conditions or groups (a condition-specific strategy), rather than across the entire dataset at once (a pooled strategy), particularly in the case of reaction times. Additionally, common procedures such as statistical transformations (e.g., log-transformation) reportedly offer at best no benefit (being instead potentially detrimental) to statistical power (Schramm & Rouder, 2019). Given the specific shape of a typical reaction distribution, treating them with bespoke models that take into account its skewness (thus reframing the notion of outliers and integrating the longer right tail of the distribution) should be considered. Examples of such models—referred to as sequential sampling models or evidence accumulation models—include Wald models (Anders et al., 2016), log-normal race models (Rouder et al., 2015), linear ballistic accumulators (Brown & Heathcote, 2008), and Drift Diffusion Models (Ratcliff et al., 2016).

Thus, when manually inspecting data for outliers, it can be helpful to think of outliers as belonging to different types of outliers, or categories, which can help decide what to do with a given outlier.

### Error, interesting, and random outliers

Several authors distinguish between error outliers, interesting outliers, and random outliers (Aguinis et al., 2013; Leys et al. 2019).[6] *Error outliers* are likely due to human error and should be corrected before data analysis or outright removed since they are invalid observations (e.g., physiologically implausible reaction times). *Interesting outliers* are not due to technical error and may be of theoretical interest; it might thus be relevant to investigate them further, even though they should be removed from the current analysis of interest. *Random outliers* are assumed to be due to chance alone and to belong to the correct distribution and, therefore, should be retained.

It is recommended to *keep* observations which are expected to be part of the distribution of interest, even if they are outliers (Leys et al., 2019). However, if it is suspected that the outliers belong to an alternative distribution, then those observations could have a large impact on the results. These observations could then call into question the robustness of these results, especially if significance is conditional on their inclusion, so they should be removed. Some authors also report detailed decision trees for handling outliers (e.g., see figures 1 and 2 in Aguinis et al., 2013).

We should also keep in mind that there might be error outliers that are not detected by statistical tools but should nonetheless be found and removed. For example, if we are studying the effects of X on Y among teenagers, and we have one observation from a 20-year-old, this observation might not be a *statistical outlier*, but it is an outlier in the

[6] Some authors provide much more detailed classifications of outliers; for example, see Table 1 in Aguinis et al. (2013), for 14 different outlier definitions based on a literature review.

*context* of our research and should be discarded. We could call these observations *undetected* error outliers, in the sense that although they do not statistically stand out, they do not belong to the theoretical or empirical distribution of interest (e.g., teenagers). In this way, we should not blindly rely on statistical outlier detection methods; doing our due diligence to investigate undetected error outliers relative to our specific research question is also essential for valid inferences.

### Winsorization

*Removing* outliers that do not belong to the distribution of interest can in this case be a valid strategy, and ideally one would report results with and without outliers to see the extent of their impact on results. This approach, however, can reduce statistical power. Therefore, some propose a *recoding* approach, namely, winsorization: bringing outliers back within acceptable limits (e.g., three MADs, Tukey & McLaughlin, 1963). However, if possible, it is recommended to collect enough data so that even after removing outliers, there is still sufficient statistical power without having to resort to winsorization (Leys et al., 2019).

The *easystats* ecosystem makes it easy to incorporate this step into your workflow through the `winsorize()` function of *{datawizard}*, a lightweight R package to facilitate data wrangling and statistical transformations (Patil et al., 2022). This procedure will bring back univariate outliers within the limits of "acceptable" values, based either on the percentile, the *z* score, or its robust alternative based on the MAD. For example, let's say we want to winsorize the univariate outlier identified before:

```
data[33:34, 2:3]  # See outliers rows

#>    mpg cyl
#> 33  12  12
#> 34  55  55

# Winsorizing using the MAD
library(datawizard)
winsorized_data <- winsorize(data, method = "zscore", robust = TRUE,
threshold = 3)

# Outlier values > +/- MAD have been winsorized
winsorized_data[33:34, 2:3]

#>         mpg     cyl
#> 33 12.00000 12.0000
#> 34 36.32403 14.8956
```

### The importance of transparency

Finally, it is a critical part of a sound outlier treatment that regardless of which SOD method used, it should be reported in a reproducible manner. Ideally, the handling of outliers should be specified *a priori* with as much detail as possible, and preregistered, to limit researchers' degrees of freedom and therefore risks of false positives (Leys et al., 2019). This is especially true given that interesting outliers and random outliers are oftentimes hard to distinguish in practice. Thus, researchers should always prioritize transparency and report all the following information: (a) how many outliers were identified (including percentage); (b) according to which method and criteria, (c) using which function of which R package (if applicable), and (d) how they were handled (excluded or winsorized, if the latter, using what threshold). If at all possible, (e) the corresponding code along with the data should be shared in a public repository like the Open Science Framework (OSF), so that the exclusion criteria can be reproduced precisely.

## Conclusion

In this paper, we have shown how to investigate outliers using the `check_outliers()` function of the *{performance}* package while following current good practices. However, best practice for outlier treatment does not stop at using appropriate statistical algorithms, but entails respecting existing recommendations, such as preregistration, reproducibility, consistency, transparency, and justification. Ideally, one would additionally also report the package, function, and threshold used (linking to the full code when possible). We hope that this paper and the accompanying `check_outliers()` function of *easystats* will help

researchers engage in good research practices while providing a smooth outlier detection experience.

**Acknowledgements** *{performance}* is part of the collaborative *easystats* ecosystem (Lüdecke et al., 2023). Thus, we thank all members of easystats, contributors, and users alike.

**Authors contributions** Writing- Original draft preparation: RT. Writing-Reviewing and Editing, Software: RT, MSB-S, IP, DL, BMW, and DM.

**Funding** This research received no external funding.

**Data availability** This paper first appeared as a preprint (https://doi.org/10.31234/osf.io/bu6nt) and is also available as an online vignette at: https://easystats.github.io/performance/articles/check_outliers. All data used in this paper uses data included with base R.

**Code availability** The performance package is available at the package official website (https://easystats.github.io/performance), on CRAN (https://cran.r-project.org/package=performance), and on the R-Universe (https://easystats.r-universe.dev/performance). The source code is available on GitHub (https://github.com/easystats/performance/), and the package can be installed from CRAN with install.packages("performance"). The code to reproduce figures and all analyses in this paper is available at https://osf.io/eqja6/.

## Declarations

**Competing interests** The authors declare no conflict of interest.

## References

Aguinis, H., Gottfredson, R. K., & Joo, H. (2013). Best-practice recommendations for defining, identifying, and handling outliers. *Organizational Research Methods, 16*(2), 270–301. https://doi.org/10.1177/1094428112470848

Anders, R., Alario, F., Van Maanen, L., et al. (2016). The shifted Wald distribution for response time data analysis. *Psychological Methods, 21*(3), 309. https://doi.org/10.1037/met0000066

Aruguete, M. S., Huynh, H., Browne, B. L., Jurs, B., Flint, E., & McCutcheon, L. E. (2019). How serious is the 'carelessness' problem on Mechanical Turk? *International Journal of Social Research Methodology, 22*(5), 441–449. https://doi.org/10.1080/13645579.2018.1563966

Brown, S. D., & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology, 57*(3), 153–178. https://doi.org/10.1016/j.cogpsych.2007.12.002

Cao, N., Lin, Y. R., Gotz, D., & Du, F. (2018). Z-Glyph: Visualizing outliers in multivariate data. *Information Visualization, 17*(1), 22–40. https://doi.org/10.1177/1473871616686635

Chaloner, K., & Brant, R. (1988). A Bayesian approach to outlier detection and residual analysis. *Biometrika, 75*(4), 651–659. https://doi.org/10.1093/biomet/75.4.651

Ciccione, L., Dehaene, G., & Dehaene, S. (2023). Outlier detection and rejection in scatterplots: Do outliers influence intuitive statistical judgments? *Journal of Experimental Psychology: Human Perception and Performance, 49*(1), 129–144. https://doi.org/10.1037/xhp0001065

Cook, R. D. (1977). Detection of influential observation in linear regression. *Technometrics, 19*(1), 15–18. https://doi.org/10.1080/00401706.1977.10489493

Curran, P. G. (2016). Methods for the detection of carelessly invalid responses in survey data. *Journal of Experimental Social Psychology, 66*, 4–19. https://doi.org/10.1016/j.jesp.2015.07.006

Gnanadesikan, R., & Kettenring, J. R. (1972). Robust estimates, residuals, and outlier detection with multiresponse data. *Biometrics, 28*(1), 81–124. https://doi.org/10.2307/2528963

Goldammer, P., Annen, H., Stöckli, P. L., & Jonas, K. (2020). Careless responding in questionnaire measures: Detection, impact, and remedies. *The Leadership Quarterly, 31*(4), 101384. https://doi.org/10.1016/j.leaqua.2020.101384

Leys, C., Ley, C., Klein, O., Bernard, P., & Licata, L. (2013). Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology, 49*(4), 764–766. https://doi.org/10.1016/j.jesp.2013.03.013

Leys, C., Klein, O., Dominicy, Y., & Ley, C. (2018). Detecting multivariate outliers: Use a robust variant of the Mahalanobis distance. *Journal of Experimental Social Psychology, 74*, 150–156. https://doi.org/10.1016/j.jesp.2017.09.011

Leys, C., Delacre, M., Mora, Y. L., Lakens, D., & Ley, C. (2019). How to classify, detect, and manage univariate and multivariate outliers, with emphasis on pre-registration. *International Review of Social Psychology*. https://doi.org/10.5334/irsp.289

Lüdecke, D., Ben-Shachar, M. S., Patil, I., Waggoner, P., & Makowski, D. (2021). performance: An R package for assessment, comparison and testing of statistical models. *Journal of Open Source Software, 6*(60), 3139. https://doi.org/10.21105/joss.03139

Lüdecke, D., Makowski, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., & Thériault, R. (2023). *easystats: Streamline model interpretation, visualization, and reporting*. R package version 0.7.0. Retrieved February 26, 2024, from https://easystats.github.io/easystats/

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and stan*. CRC Press.

McNeil, D. R. (1977). *Interactive Data Analysis: A Practical Primer*. Wiley.

Miller, J. (2023). Outlier exclusion procedures for reaction time analysis: The cures are generally worse than the disease. *Journal of Experimental Psychology: General*. https://doi.org/10.1037/xge0001450

Patil, I., Makowski, D., Ben-Shachar, M. S., Wiernik, B. M., Bacher, E., & Lüdecke, D. (2022). datawizard: An R package for easy data preparation and statistical transformations. *Journal of Open Source Software, 7*(78), 4684. https://doi.org/10.21105/joss.04684

Ratcliff, R. (1993). Methods for dealing with reaction time outliers. *Psychological Bulletin, 114*(3), 510. https://doi.org/10.1037/0033-2909.114.3.510

Ratcliff, R., Smith, P. L., Brown, S. D., & McKoon, G. (2016). Diffusion decision model: Current issues and history. *Trends in Cognitive Sciences, 20*(4), 260–281. https://doi.org/10.1016/j.tics.2016.01.007

Rouder, J. N., Province, J. M., Morey, R. D., Gomez, P., & Heathcote, A. (2015). The lognormal race: A cognitive-process model of choice and latency with desirable psychometric properties. *Psychometrika, 80*, 491–513. https://doi.org/10.1007/s11336-013-9396-3

Schramm, P., & Rouder, J. N. (2019). Are reaction time transformations really beneficial? *PsyArXiv*. https://doi.org/10.31234/osf.io/9ksa6

Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science, 22*(11), 1359–1366. https://doi.org/10.1177/0956797611417632

Smiti, A. (2020). A critical overview of outlier detection methods. *Computer Science Review, 38*, 100306. https://doi.org/10.1016/j.cosrev.2020.100306

Tukey, J. W., & McLaughlin, D. H. (1963). Less vulnerable confidence and significance procedures for location based on a single sample:

Trimming/winsorization 1. *Sankhyā: The Indian Journal of Statistics, Series A*, 331–352.

Van Zandt, T., & Ratcliff, R. (1995). Statistical mimicking of reaction time data: Single-process models, parameter variability, and mixtures. *Psychonomic Bulletin & Review, 2*(1), 20–54. https://doi.org/10.3758/BF03214411

Ward, M. K., & Meade, A. W. (2023). Dealing with careless responding in survey data: Prevention, identification, and recommended best practices. *Annual Review of Psychology, 74*(1), 577–596. https://doi.org/10.1146/annurev-psych-040422-045007

Yentes R.D., & Wilhelm, F. (2023). *careless: Procedures for computing indices of careless responding*. R package version 1.2.2. Retrieved February 26, 2024, from https://cran.r-project.org/package=careless

Zijlstra, W. P., van der Ark, L. A., & Sijtsma, K. (2011). Outliers in questionnaire data: Can they be detected and should they be removed? *Journal of Educational and Behavioral Statistics, 36*(2), 186–212. https://doi.org/10.3102/1076998610366263