# Contamination models in the R package simFrame for statistical simulation

A. Alfons[1], M. Templ[1,2], P. Filzmoser[1]
[1]*Department of Statistics and Probability Theory, Vienna University of Technology*
[2]*Methods Unit, Statistics Austria*
*Vienna, Austria*
e-mail: alfons@statistik.tuwien.ac.at

**Abstract**

Due to the complexity of robust statistical methods, simulation studies are widely used to gain insight into the quality of these procedures. The R package simFrame is an object-oriented framework for statistical simulation with special emphasis on applications in robust statistics. Contamination is thereby modeled as a two-step process. Furthermore, the existing framework may be extended with user-defined contamination models.

## 1 Introduction

Robust statistical methods are becoming increasingly complex, therefore obtaining analytical results about their properties is becoming more and more time-consuming and difficult—often virtually impossible. On the other hand, computers are ever getting faster and cheaper. Therefore simulation studies are widely used by researchers to gain insight into the quality of the developed methods in different situations.

Two main concepts for simulation studies are distinguished in the literature: *model-based* and *design-based* simulation. In model-based simulation, data are generated repeatedly from a certain distribution. In every iteration, different methods are applied and quantities of interest are computed for comparison. Reference values can be obtained from the underlying theoretical distribution where appropriate. In design-based simulation, samples are drawn repeatedly from a finite population. Since real population data is only in few cases available to researchers, synthetic populations need to be generated [2]. In every iteration, certain estimators such as indicators are computed. Where appropriate, these can be compared to the true population values.

When investigating robust methods, outliers need to be included. For model-based simulation, reference values are then computed from the theoretical distribution of the non-contaminated data. For design-based simulation, the situation is more complex [1]. The most realistic scenario would be to include outliers in the population data. However, total control over the amount of contamination is required for proper evaluation of robust methods. It is therefore suggested to generate outliers in the samples. In any case, reference values are computed from the non-contaminated population values.

The R package simFrame [3] is a general framework for simulation studies in statistics. Its object-oriented implementation provides clear interfaces for extensions by the user. One of the main advantages of simFrame is that simulation studies can be defined in terms of *control objects*. For large research projects, this ensures that results obtained by different partners are comparable.

## 2 Contamination models in `simFrame`

In the literature on robust statistics, the distribution $F$ of contaminated data is typically modeled as a mixture of distributions

$$F = (1 - \varepsilon)G + \varepsilon H, \tag{1}$$

where $\varepsilon$ denotes the *contamination level*, $G$ is the distribution of the non-contaminated part of the data and $H$ is the distribution of the contamination [5]. As a consequence, outliers may be modeled by a two-step process [4]. The first step is to select observations to be contaminated, the second is to model the distribution of the outliers. Let $n$ be the number of observations, $p$ the number of variables, and let $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$, $i = 1, \ldots, n$, denote the observations.

1. Let $O_i$, $i = 1, \ldots, n$, be an indicator whether an observation is an outlier ($O_i = 1$) or not ($O_i = 0$). The situation that the probability distribution of $O_i$ does not depend on any other variables, i.e., that

   $$P(O_i = 1 | x_{i1}, \ldots, x_{ip}) = P(O_i = 1), \qquad i = 1, \ldots, n \tag{2}$$

   may be called *outlying completely at random* (OCAR). If Equation (2) is violated, i.e., if the probability distribution of $O_i$ depends on observed information, the situation may be called *outlying at random* (OAR).

2. Let $I_c := \{i = 1, \ldots, n : O_i = 1\}$ be the index set of the observations to be contaminated, and let $\boldsymbol{x}_i^* = (x_{i1}^*, \ldots, x_{ip}^*)$ denote the true (i.e., non-contaminated) values of $\boldsymbol{x}_i$, $i \in I_c$. If the distribution $H$ does not depend on the true values, i.e., if $\boldsymbol{x}_i \sim H(x_1, \ldots, x_p)$, $i \in I_c$, the outliers may be called *contaminated completely at random* (CCAR). On the other hand, if $H$ depends on the true values, i.e., if $\boldsymbol{x}_i \sim H(x_1, \ldots, x_p, x_{i1}^*, \ldots, x_{ip}^*)$, $i \in I_c$, the outliers may be called *contaminated at random* (CAR).

The package `simFrame` is implemented in the open-source statistical environment and programming language `R` [6]. Taking advantage of object-oriented programming, the control classes `DCARContControl` and `DARContControl` determine how contamination is handled in simulation studies (see the example in Section 3). `DCARContControl` may be used for OCAR-CCAR and OAR-CCAR models, whereas `DARContControl` corresponds to OCAR-CAR and OAR-CAR. Additional contamination models may be added in the future. However, the object-oriented design further allows contamination models to be implemented by the user. The programming interfaces for such extensions are described in detail in [3].

## 3 Example: outlier detection

This simple motivational example for the usage of `simFrame` is a comparison of outlier detection using classical and robust estimation of location and scatter. The robust estimates are obtained with the fast MCD [7] implementation in package `rrcov` [8].

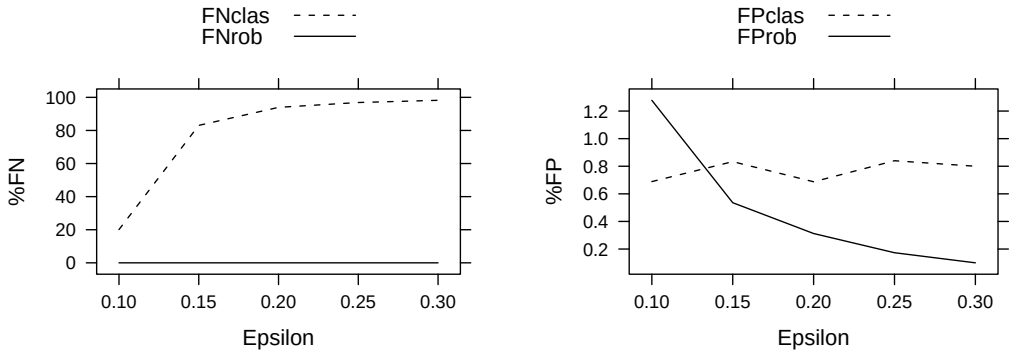


Figure 1: Average proportions of false negatives (*left*) and false positives (*right*).

Data are generated in each of the 100 simulation runs from a two-dimensional normal distribution. Varying the contamination level between 10% and 30% in steps of 5%, the contaminated data are generated from a normal distribution with a shifted mean (OCAR-CCAR). In the function to be executed in every iteration, the percentages of false negatives and false positives are computed. Note that the default tuning parameter 0.5 is used for the MCD. For reproducibility of the simulation results, the seed of the random number generator is set before running the simulation study.

```
> sigma <- matrix(c(1, 0.5, 0.5, 1), 2, 2)
> dc <- DataControl(size = 100, distribution = rmvnorm, dots = list(sigma = sigma))
> cc <- DCARContControl(epsilon = seq(0.1, 0.3, by = 0.05), distribution = rmvnorm,
+     dots = list(mean = c(5, -5), sigma = sigma))
> sim <- function(x, q) {
+     clas <- Cov(x[, 1:2])
+     rob <- CovMcd(x[, 1:2])
+     dclas <- mahalanobis(x[, 1:2], clas@center, clas@cov)
+     drob <- mahalanobis(x[, 1:2], rob@center, rob@cov)
+     outclas <- dclas > q
+     outrob <- drob > q
+     nout <- length(which(x$.contaminated))
+     ngood <- nrow(x) - nout
+     c(FNclas = length(which(!outclas & x$.contaminated))/nout,
+         FNrob = length(which(!outrob & x$.contaminated))/nout,
+         FPclas = length(which(outclas & !x$.contaminated))/ngood,
+         FProb = length(which(outrob & !x$.contaminated))/ngood) *
+         100
+ }
> set.seed(12345)
> result <- runSimulation(dc, nrep = 100, contControl = cc, fun = sim,
+     q = qchisq(0.975, df = 2))
> plot(result, select = c("FNclas", "FNrob"), ylab = "%FN")
> plot(result, select = c("FPclas", "FProb"), ylab = "%FP")
```

In `simFrame`, a suitable graphical representation of the results is selected automatically depending on their structure. Figure 1 shows plots of the average proportions of

false negatives (*left*) and false positives (*right*). The plots, of course, clearly favor the MCD over classical estimation.

# 4 Conclusions

The package `simFrame` is an object-oriented framework for simulation studies in the statistical environment R. Different contamination models are implemented using control classes. The flexible framework further allows additional contamination models to be implemented by the user. Hence `simFrame` is widely applicable in the field of robust statistics.

# Acknowledgments

This work was partly funded by the European Union within the $7^{\text{th}}$ framework programme for research (Project AMELI, Grant Agreement No. 217322).

# References

[1] Alfons A., Templ M., Filzmoser P., Kraft S., Hulliger B (2009). Intermediate report on the data generation mechanism and on the design of the simulation study. *AMELI Deliverable 6.1*, Department of Statistics and Probability Theory, Vienna University of Technology.

[2] Alfons A., Kraft S., Templ M., Filzmoser P. (2010). Simulation of synthetic population data for household surveys with application to EU-SILC. *Research Report CS-2010-1*, Department of Statistics and Probability Theory, Vienna University of Technology.

[3] Alfons A., Templ M., Filzmoser P. (2009). `simFrame`: An object-oriented framework for statistical simulation. *Research Report CS-2009-1*, Department of Statistics and Probability Theory, Vienna University of Technology.

[4] Hulliger B., Schoch T. (2009). Robust multivariate imputation with survey data. *$57^{th}$ Session of the International Statistical Institute*, Durban.

[5] Maronna R., Martin D., Yohai V. (2006). *Robust Statistics*. Wiley, Chichester.

[6] R Development Core Team (2009). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna.

[7] Rousseeuw P.J., Van Driessen K. (1999). A fast algorithm for the minimum covariance determinant estimator. *Technometrics*. Vol. **41**(3), pp. 212–223.

[8] Todorov V., Filzmoser P. (2009). An object-oriented framework for robust multivariate analysis. *Journal of Statistical Software*. Vol. **32**(3), pp. 1–47.