The Journal of Open Source Software

experDesign: stratifying samples into batches with minimal bias

Lluís Revilla Sancho[1, 2], **Juan-José Lozano**[1], **and Azucena Salas**[2]

1 Centro de Investigación Biomédica en Red, Enfermedades Hepáticas y Digestivas **2** Institut d'Investigacions Biomèdiques August Pi i Sunyer, IDIBAPS

DOI: 10.21105/joss.03358

Software

- Review
- Repository
- Archive

Editor: Lorena Pantano

Reviewers:

- @abartlett004
- @stemangiola

Submitted: 23 April 2021
Published: 27 November 2021

Summary

The design of an experiment is critical to its success. Nonetheless, even when correctly designed, the process leading up to the moment of measuring a given variable is critical. At any one of the several steps, from sample collection to measurement of a variable, various errors and problems can affect the experimental results. Failure to take such variability into account can render an experiment inconclusive. *experDesign* provides tools to minimize the risk of inconclusive results by assigning samples to batches to reduce potential batch effects.

Introduction

To design an experiment that can support conclusive results upon analysis, the source of the variation between samples must be identified. Typically, one can control the environment in which the study or experiment is being conducted. Sometimes, however, this is not possible. In such cases, techniques to control variations must to be applied. There are three methods used to decrease the uncertainty of the unwanted variation: blocking, randomization and replication (Klaus 2015).

Blocking is a method that groups samples that are equal according to one or more variables, allowing the estimation of the differences between each batch by comparing measurements within the blocks. **Randomization** minimizes the variation in the measurements by randomly mixing the potential confounding variables. **Replication** increases the number of samples used in an experiment to better estimate the variation of the experiment. In some settings these techniques can be applied together to enhance the robustness of the study.

Between the designing of an experiment and the measurement of the samples, some samples might be lost, contaminated, or degraded below the quality threshold. In addition, experiments will occasionally need to be carried out in batches. The later might be needed for technical reasons; for example, the device cannot measure more than a given number of samples at the same time. Practical reasons can also be a factor; for instance, it may not be possible to obtain additional measurements in the field during the allotted time.

This divergence from the original design might cause batch effects, thereby perturbing the analysis. There are several techniques to identify and assess batch effects when analyzing an already measured experiment (Leek et al. 2010). It would be better to avoid such batch effects before executing an experiment. By taking into account the differences between the original design and the state before the measurement is conducted, confounding effects can be minimized.

To prevent the batch effect from confounding the analysis after the initial design of the experiment, there are two options: randomization and replication. Randomization, consists of

shuffling the samples in order to mix different attributes, which can help reduce variations across groups. In contrast, replication helps estimate the variation of the measurements or samples, thus increasing the precision of the estimates of the true value obtained by the analysis. Replications consist of increasing the number of measurements with similar attributes. When a sample is measured multiple times, this is referred to as a technical replicate. Technical replicates help estimate the variation of the measurement method, and thus the possible batch effect (Blainey, Krzywinski, and Altman 2014).

Randomization and replication can be used to prevent batch effects that might confound the analysis. By examining how the variables are distributed across each batch, proper randomization can be ensured, thus minimizing batch effects. This is known as randomized block experimental design or stratified random sampling experimental design.

State of the art

There are certain tools that can minimize batch effects on the R language in multiple fields, particularly for biological research (R Core Team 2014). Here we briefly describe the currently available packages:

- *OSAT*, at Bioconductor, first allocates the samples from each batch according to a variable; it then shuffles the samples from each batch in order to randomize the other variables (Yan et al. 2012). This algorithm relies on categorical variables and cannot use numerical variables (e.g., those that are age- or time-related) unless they are treated as categorical variables.
  OSAT provides templates for plates that hold 2, 4, 8 Illumina BeadChip chips, having 24, 48 or 96 wells. Moreover, it works for both numeric and categorical variables but OSAT might return less rows than the input provided because they might have NA value.

- *anticlust*, at CRAN, divides the samples into similar groups, ensuring similarity by enforcing heterogeneity within groups (Papenberg and Klau 2020). Conceptually it is similar to the clustering method k-means.
  anticlust does not handle all types of variables, it only accepts numeric variables.

- Recently, *Omixer*, a new package, has been made available at Bioconductor (Sinke, Cats, and Heijmans 2021). It tests whether the random assignments made by it are homogeneous by transforming all variables to numeric values and using the Kendall's correlation when there are more than 5 samples; otherwise, it utilizes the Pearson's chi-squared test.
  There is a bug in the Omixer that prevents it from working unless specific conditions are meet. This precluded any comparisons of Omixer with other tools using the same settings.

For completeness a description and comparison of the usage of the different software packages currently available on CRAN and Bioconductor is presented below. First, we start with some real data obtained from a survey. This data set has three variables of interest; Sex, Smoke and Age are a mix of categorical and numeric variables.

Statement of need

Current solutions for stratifying samples to reduce and control batch effect do not work for all cases. They are either specialized to a particular type of data, they omit some conditions that

are usually met, or they only work under a specific subset of conditions. The new package ***experDesign*** works with all data types and does not require a spatial distribution making it suitable for all kind of experiments. This package is intended for people needing a quick and easy solution that will provide reasonable suggestions on how to best distribute the samples for analysis.

Description

The package ***experDesign*** provides the function `design` to arrange the samples into multiple batches such that a variable's distribution remains homogeneous within each batch. Each batch is set to have some centrality and dispersion statistics to match as closely as possible with the original input design data. The statistics used are the mean, the standard deviation, the median absolute deviation, variables with no value number, the entropy and the independence of the categorical variables. With each iteration if the random distribution of the sample statistics for each batch has fewer differences vis-à-vis the original distribution than the last stored sample distribution then it replaces it as the best sample distribution. Upon completion of the iterations the best sample distribution is returned to the user.

Users can examine the following flowchart to decide what function(s) they need to use:

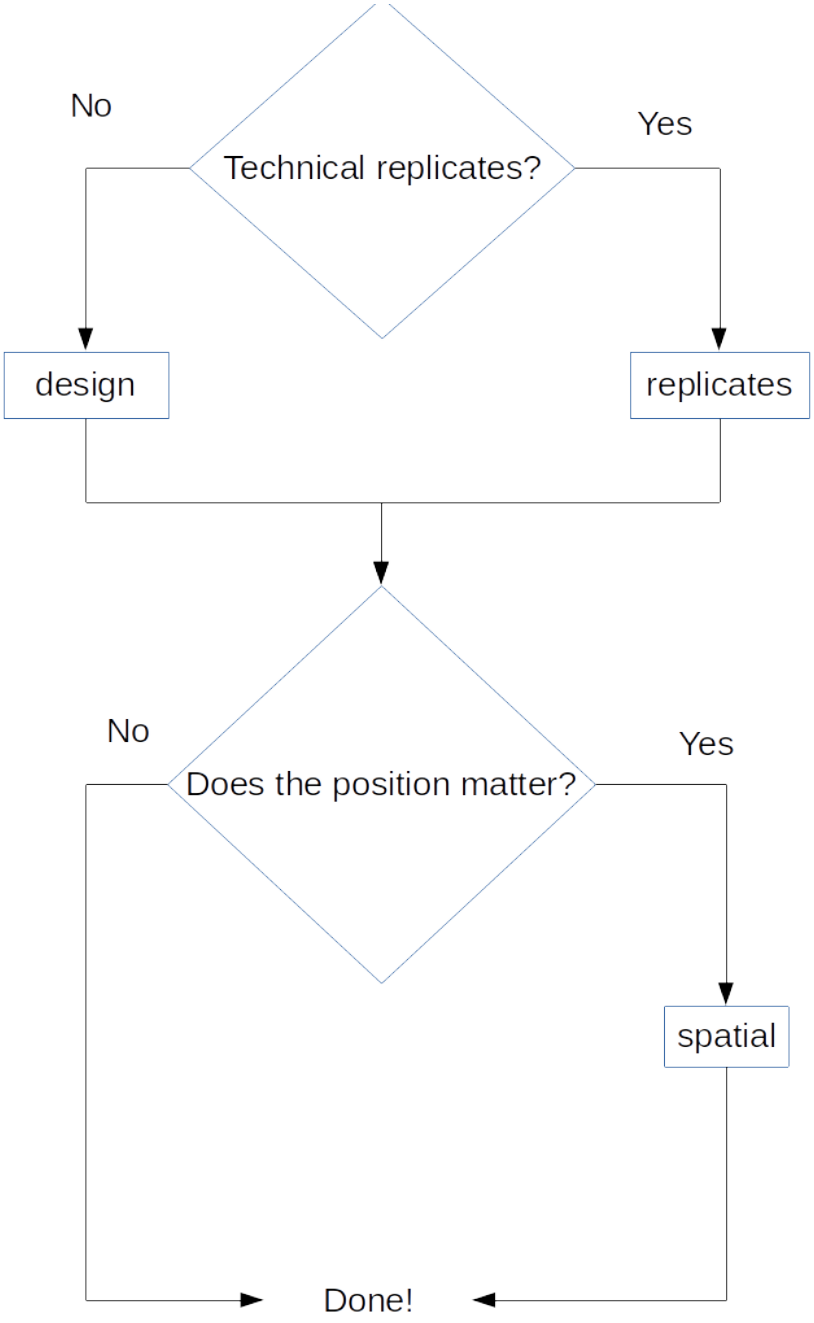

Figure 1: Flow chart to decide which functions are needed

If users want a design without replicates but the batches have some spatial distribution, we must use `design` to allocate the samples on each batch, followed by the `spatial` function to randomly distribute the samples homogeneously by position within each batch. See the example in `inspect` and the vignette:

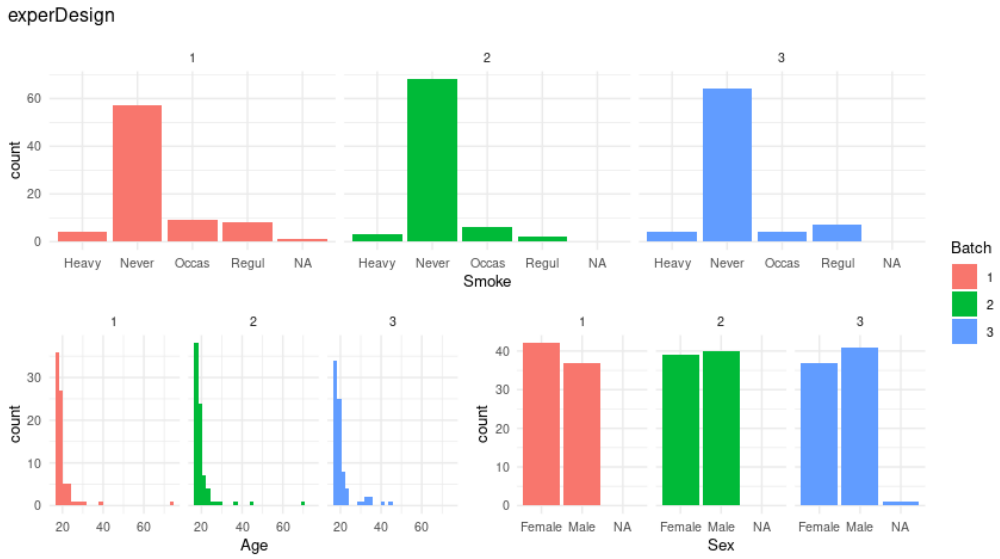

Figure 2: Example of distributions generated with ***experDesign***.

On Figure 2 one can see that the distribution index generated by experDesign uses both numeric and categorical variables keeping also the samples with missing (NA) information.

The statistics of the index can be checked for multiple statistics, as shown on the help pages of `evaluate_na`, `evaluate_entropy`, `evaluate_mad`, `evaluate_sd` and `evaluate_mean`. We can also compare our results with the original distribution via `evaluate_orig`.

In addition to distributing the samples into batches, ***experDesign*** provides tools to add technical replicates. In order to choose them from the available samples, the function `extreme_cases` is provided. For easier usage, the `replicates` function designs an experiment with the desired number of replicates per batch.

experDesign also provides several small utilities to make it easier to design the experiment in batches. For instance, a function called `sizes_batches` helps calculate the number of samples in order to distribute them across the required batches. Furthermore, `optimum_batches` calculates the minimal number of batches required. Examples of all this methods can be found on the manual page of each function and on the vignette.

In conclusion ***experDesign*** offers a fast method for preparing a batched experiment. It can use as many numeric and categorical variables as needed to stratify the experimental design based on batches including spatial distributions.

Acknowledgments

We are grateful to Joe Moore for English-language assistance and we would like to thank reviewers and the editor for taking the time and effort necessary to review the manuscript.

References

Blainey, Paul, Martin Krzywinski, and Naomi Altman. 2014. "Replication." *Nature Methods* 11 (9): 879–80. https://doi.org/10.1038/nmeth.3091.

Klaus, Bernd. 2015. "Statistical Relevancerelevant Statistics, Part i." *The EMBO Journal* 34 (22): 2727–30. https://doi.org/10.15252/embj.201592958.

Leek, Jeffrey T., Robert B. Scharpf, Héctor Corrada Bravo, David Simcha, Benjamin Langmead, W. Evan Johnson, Donald Geman, Keith Baggerly, and Rafael A. Irizarry. 2010. "Tackling the Widespread and Critical Impact of Batch Effects in High-Throughput Data." *Nature Reviews. Genetics* 11 (10). https://doi.org/10.1038/nrg2825.

Papenberg, Martin, and Gunnar W. Klau. 2020. "Using Anticlustering to Partition Data Sets into Equivalent Parts." *Psychological Methods*. https://doi.org/10.1037/met0000301.

R Core Team. 2014. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://R-project.org/.

Sinke, Lucy, Davy Cats, and Bastiaan T Heijmans. 2021. "Omixer: Multivariate and Reproducible Sample Randomization to Proactively Counter Batch Effects in Omics Studies." *Bioinformatics*, no. btab159 (March). https://doi.org/10.1093/bioinformatics/btab159.

Yan, Li, Changxing Ma, Dan Wang, Qiang Hu, Maochun Qin, Jeffrey M. Conroy, Lara E. Sucheston, et al. 2012. "OSAT: A Tool for Sample-to-Batch Allocations in Genomics Experiments." *BMC Genomics* 13 (1): 689. https://doi.org/10.1186/1471-2164-13-689.