*lancer*: an R package for linearity assessment and visualisation of multiple curves

## Manuscript Type

Application Note

# Abstract

## Summary

Linearity assessment plays a significant role in the validation of instrumentation and experimental procedures. Linearity can be tested by applying several graphical and numerical approaches. Commonly used spreadsheet software, e.g. Microsoft Excel, only allow the analyst to repetitively plot, view and analyse the linearity of curves one at a time, a tedious and time-consuming process. In addition, summary statistics of these curves are mostly limited to the Pearson Correlation Coefficient, which is insufficient to fully test the for linearity. While plotting of many curves at once and calculation of additional summary statistics for assessing linearity of curves can be done using programming languages like R, implementing this from scratch can be arduous task for the analyst. As such, we created the R package *lancer* so that curves can be analysed efficiently with a few functions. To our knowledge, this is currently not implemented in other software or programming languages. *lancer* furthermore reports the statistical results in tables/spreadsheets and records the plots in a pdf file. In addition, *lancer* can also create interactive trellis plots, displayed as a HTML folder, for exploratory analysis, helping analysts when dealing with large datasets.

## Availability and implementation

*lancer* is available on GitHub <https://github.com/SLINGhub/lancer>. The documentation and tutorials can be accessed from <https://slinghub.github.io/lancer/>

## Supplementary information

Supplementary data are available at *Bioinformatics* online.

## Issue Section

Data and text mining

# Introduction

Linearity assessment is an important quality control to verify if the quantitative acquisition range of an instrument or experimental method is reliable. It is applied in many fields in analytical science such as calibration/dilution studies (Rodríguez *et al.* (1993) and Sands *et al.* (2021)) and assay development (Ross and Sweep (2003) and Hsieh and Liu (2008)). After analysis, curves are plotted individually for each analyte of interest with a Pearson Correlation Coefficient value for each plotted curve using general-purpose software like Excel. A non-linear curve suggests possible human, systematic or analytical errors that need to be addressed. Examples of these errors are incorrect amount of analyte, imprecise pipette calibration, expired reagents or equipment malfunction.

However, as instrument or experimental method are becoming more high throughput, many measurements can be done in a short time. Having the analyst to individually plot numerous curves to check for linearity is time-consuming. Furthermore, Sonnergaard (2006) warns that the Pearson Correlation Coefficient is an ineffective standalone numeric parameter to estimate linearity. While researchers have created other metrics for linearity evaluation, these metrics are rarely implemented in most general-purpose software.

The R package, *lancer* addresses these issues by assisting analysts, to plot curves from many experiments easily with additional metrics, other than the Pearson Correlation Coefficient, that better describe the curve’s shape. It also provides an interactive viewer for analysts to group, filter and sort the curves, allowing them to examine problematic ones, such as saturated curves.

# Approach

Using response curves in metabolomic/lipidomic studies as an example, Supplementary Figure 1 depicts the workflow of *lancer*. The workflow starts with two tables:

* Curve Signal Data, containing response signals (y-axis for response curve) for each sample and curve.
* Curve Batch Annotation, containing curve related information, such as concentration (x-axis for response curve) and response curve batches if any.

Using a common column Sample Name, the two tables can be merged into one table (Curve Table) via create\_curve\_table.

Next, summary statistics are calculated via summarise\_curve\_table for each curve. Besides the Pearson Correlation Coefficient, one additional calculation is the Mandel’s Fitting Test ( in [Equation 1](#eq-mandel-test)) described by Andrade and Gómez-Carracedo (2013). A low value from the test gives sufficient evidence that a quadratic model fits better than a linear model, indicating a non-linear curve.

Another calculation is Percent Residual Accuracy ( in [Equation 2](#eq-pra)) from Logue and Manandhar (2018). Ranging from to , a linear curve gives a value close to .

The software also calculates the concavity of the fitted quadratic model to identify if the curve is dominantly non-linear at high (concavity ) or low (concavity ) concentrations.

Supplementary Figure 2 gives the summary statistics of three manually generated curves: A linear curve and two curves with a plateau at high concentrations (denoted as saturation regime curve) and low concentrations (denoted as noise regime curve) respectively. The corresponding Pearson Correlation Coefficient values (r\_corr) are (, and respectively), even for the non-linear curves. However, both non-linear curves give a much lower Mandel’s Fitting Test values (mandel\_p\_val) ( and respectively vs ). Likewise, the Percent Residual Accuracy values (pra\_linear) are much lower in the non-linear curves compared to the linear curve ( and respectively vs ).

|  |  |  |
| --- | --- | --- |
| |  | | --- | | (A) Curve Grouping Workflow | |  |

|  |  |
| --- | --- |
|  |  |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | (B) Interactive Visualisation | |  |

Fig. 1. *lancer*’s curve grouping workflow in [Fig. 1 (A)](#fig-linearity) and interactive visualisation of curves in [Fig. 1 (B)](#fig-visualisation).

After calculating the summary statistics for each curve, *lancer* uses the function evaluate\_linearity to group the curves according to the workflows proposed in [Fig. 1 (A)](#fig-linearity). Workflow 1 uses the Pearson Correlation Coefficient and Percent Residual Accuracy to determine if the curve is linear (labelled as Good Linearity) or not (labelled as Poor Linearity). Workflow 2 goes one step further, using the Mandel’s Fitting Test and the fitted quadratic model’s concavity to check if the non-linear curve plateaus at low (labeled as limit of detection) or high (labelled as saturation) concentrations. Non-linear curves that do not follow these trends are labelled as Poor Linearity.

A benchmark workflow using only Pearson Correlation Coefficient value of is compared with Workflow 2 on simulated data sets of 200 linear curves (labelled as Linear), curves that plateau at low (labelled as Noise Regime) or high (labelled as Saturation Regime) concentrations. Supplementary Figure 3 shows that Workflow 2 better identifies the saturation and noise regime curves than the benchmark workflow. While Workflow 2 correctly classifies less linear curves than the benchmark workflow, its percentage of correctly classified linear curves 181/200 (90.5%) is high. See <https://lancer-simulation.netlify.app> for report details. While the threshold values of Pearson Correlation Coefficient and Percent Residual Accuracy are based on the interpretation of Y. H. Chan (2003) and Logue and Manandhar (2018), respectively, they remain subjective and arbitrary. Nevertheless, *lancer* allows optimization of these threshold values according to the analyst’s preference.

Although *lancer* can export the results in Excel or pdf, they may be too complex for meaningful interpretation. [Fig. 1 (B)](#fig-visualisation) shows a HTML folder, exported by *lancer*, such that clicking on the index.html file inside the folder will open an interactive trellis plot that can group, filter and sort curves. This allows room for exploratory data analysis, such as identifying curves with linearity issues or understanding the effects of changing the Pearson Correlation Coefficient threshold to another value. Such information is hard to achieve with the Excel and pdf files. An example of an interactive viewer created by *lancer* can be viewed at <https://lancer-interactive-example.netlify.app>

# Conclusion

To check if an instrument or experimental method is reliable, it is crucial to check for linearity. However, there are few software that can do this efficiently at a high throughput setting. R package, *lancer*, rectifies this with functions that plots many curves quickly and reports alternative curve summary statistics, other than the Pearson Correlation Coefficient, to better describe the shape of curves. It also provides an interactive trellis plot for exploratory data analysis. It is available on GitHub <https://github.com/SLINGhub/lancer> while the documentation and tutorials can be accessed from <https://slinghub.github.io/lancer>.

# Acknowledgements

These should be included at the end of the text and not in footnotes. Please ensure you acknowledge all sources of funding, see funding section below.

Details of all funding sources for the work in question should be given in a separate section entitled ‘Funding’. This should appear before the ‘Acknowledgements’ section.

# Funding

The following rules should be followed:

* The sentence should begin: ‘This work was supported by …’ -
* The full official funding agency name should be given, i.e. ‘National Institutes of Health’, not ‘NIH’ (full RIN-approved list of UK funding agencies)
* Grant numbers should be given in brackets as follows: ‘[grant number xxxx]’
* Multiple grant numbers should be separated by a comma as follows: ‘[grant numbers xxxx, yyyy]’
* Agencies should be separated by a semi-colon (plus ‘and’ before the last funding agency)
* Where individuals need to be specified for certain sources of funding the following text should be added after the relevant agency or grant number ‘to [author initials]’.

An example is given here: ‘This work was supported by the National Institutes of Health [AA123456 to C.S., BB765432 to M.H.]; and the Alcohol & Education Research Council [hfygr667789].’

Oxford Journals will deposit all NIH-funded articles in PubMed Central. See Depositing articles in repositories – information for authors for details. Authors must ensure that manuscripts are clearly indicated as NIH-funded using the guidelines above.

# References

Andrade,J.M. and Gómez-Carracedo,M.P. (2013) [Notes on the use of Mandel’s test to check for nonlinearity in laboratory calibrations](https://doi.org/10.1039/c2ay26400e). *Analytical Methods*, **5**, 1145.

Hsieh,E. and Liu,J. (2008) [On Statistical Evaluation of the Linearity in Assay Validation](https://doi.org/10.1080/10543400802071378). *Journal of Biopharmaceutical Statistics*, **18**, 677–690.

Logue,B.A. and Manandhar,E. (2018) [Percent residual accuracy for quantifying goodness-of-fit of linear calibration curves](https://doi.org/10.1016/j.talanta.2018.07.046). *Talanta*, **189**, 527–533.

Rodríguez,L.C. *et al.* (1993) [Estimation of Performance Characteristics of an Analytical Method Using the Data Set Of The Calibration Experiment](https://doi.org/10.1080/00032719308019900). *Analytical Letters*, **26**, 1243–1258.

Ross,H.A. and Sweep,C.G.J. (2003) [An improved procedure for testing for assay linearity](https://doi.org/10.1258/000456303321016204). *Annals of Clinical Biochemistry: International Journal of Laboratory Medicine*, **40**, 75–78.

Sands,C.J. *et al.* (2021) [Representing the metabolome with high fidelity: Range and response as quality control factors in LC-MS-based global profiling](https://doi.org/10.1021/acs.analchem.0c03848). *Analytical Chemistry*, **93**, 1924–1933.

Sonnergaard,J.M. (2006) [On the misinterpretation of the correlation coefficient in pharmaceutical sciences](https://doi.org/10.1016/j.ijpharm.2006.06.001). *International Journal of Pharmaceutics*, **321**, 12–17.

Y. H. Chan (2003) [Biostatistics 104: Correlational analysis](http://www.smj.org.sg/article/biostatistics-104-correlational-analysis). *Singapore Medical Journal*, **44**, 614–619.