Supporting Information to

Reaction Chemistry & Engineering's

Extracting kinetic information in catalysis: an automated tool for the exploration of small data

Pedro S. F. Mendes†,*, Sébastien Siradze†, Laura Pirro, Joris W. Thybaut*

Laboratory for Chemical Technology, Department of Materials, Textiles and Chemical Engineering, Ghent University, Technologiepark 125, 9052 Ghent, Belgium

†These authors contributed equally.

*Corresponding authors: Pedro.Mendes@UGent.be; Joris.Thybaut@UGent.be

Contents

Appendix A: Data handling

Mass balance check

The molar flow rates are first converted to mass flow rates using the molar masses. For each measurement, Eq. S1, where the mass flowrates at the reactor outlet, $F_{m,i}$, are compared to the mass flowrates at the reactor inlet, $F°_{m,i}$, closure of the mass balance must be satisfied to ensure that the measurements are not irredeemably affected by experimental error. If the mass balance is not satisfied within a given percentage (10 % by default), this is indicated by the tool with a warning.

$$\frac{\left|\sum_i F°_{m,i} - \sum_i F_{m,i}\right|}{\sum_i F°_{m,i}} < 0.01 \qquad (Eq. S1)$$

Calculation of meaningful variables

Together with temperature, total pressure and partial pressures of the reactants, the space-time represents a crucial independent variable in the construction and validation of a kinetic model and it is here calculated via Eq. S2:

$$Space\text{-}time = W/F°(R_1) \qquad (Eq. S2)$$

Where W indicates the mass of catalyst loaded in the reactor and $F°(R_1)$ is the inlet molar flowrate of the limiting reactant.

To decouple from the effect of total pressure, the partial pressures of the components of interest are better expressed as a ratio of partial pressures (in this case, normalized by the limiting reactant), as calculated via Eq. S3:

$$p(R_i)/p(R_1) = F°(R_i)/F°(R_1) \qquad (Eq. S3)$$

These independent variables are meant to represent the various degrees of freedom of the reaction in a meaningful manner.

Conversion and selectivities are commonly used to measure the performance of a catalyst and are therefore meaningful dependent variables. Those are calculated using Eq. S4 and Eq. S5 respectively:

$$\frac{\left|\sum_i F^{\circ}{}_{m,i} - \sum_i F_{m,i}\right|}{\sum_i F^{\circ}{}_{m,i}} < 0.01 \qquad (Eq.S4)$$

$$S(P_i) = \frac{F(P_i) - F^{\circ}(P_i)}{F^{\circ}(R_1) - F(R_1)} \cdot \frac{\#C(P_i\, from\, R_1)}{\#C(R_1)} \qquad (Eq.S5)$$

Where $\#C$ represents the number of carbon atoms in a specific species, either the limiting reactant R_1 or the product of interest P_i.

As the values for the independent variables can sometimes show slight variations, a margin of error is requested from the user for each independent variable. The code then checks if any values for the independent variables are within this margin of error and replaces the values by their average if they are. This is important during the generation of the sub-datasets, which is discussed in Section 3.2.2, as the code needs to knows when variables can be considered to be constant.

Appendix B: Smoothing factor

The recommended value for the smoothing factor of a spline in the UnivariateSpline algorithm equals the number of data points m. The effect of the smoothing factor on the generated spline was tested on fictional datasets with a small number of data points (10 or lower). Three examples of the results obtained by varying the smoothing factor are shown in Figure S1. It can be seen that for the first dataset, the recommended smoothing factor leads to a chemically intuitive result, while the lower smoothing factors lead to chemically unrealistic results. For the two other datasets, it is clear that the default smoothing factor leads to a spline which is unable to capture the linear trends in the data. In these cases, a lower smoothing factor is needed to lead to a chemically intuitive curve. Based on these results, it can be concluded that it is not possible to determine a single smoothing factor which leads to a realistic curve for all datasets.

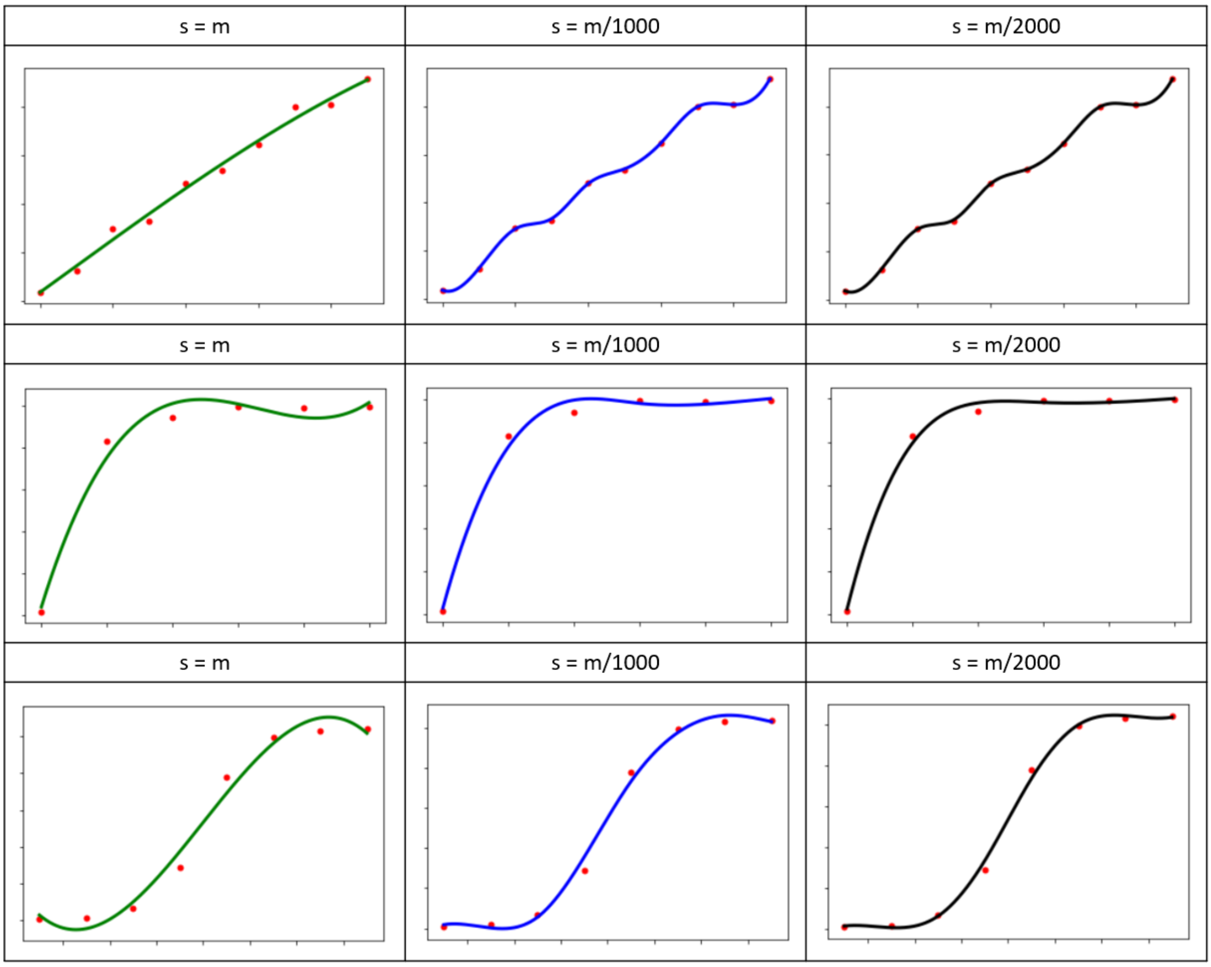


Figure S1: Splines generated by the UnivariateSpline algorithm for three fictional datasets using a smoothing factor of m, m/1000 and m/2000.

Appendix C: Maximal number of knots

To limit the number of splines which can be generated for a dataset, a maximal number of knots is used for the splines. Multiple tests were therefore done to see how many knots are needed to be able to extract all trends which can realistically be expected from kinetic data. In Figure S2, splines with different numbers of knots are shown for a dataset representing an S-curve trend, showcasing that 7 knots are sufficient to generate a curve which is visually almost identical to an actual S-curve. This trend was chosen as it contains two linear sections, making it particularly difficult to represent using a cubic spline with a low number of knots. All curves in Figure S1 in Appendix A were also generated with less than 7 knots.

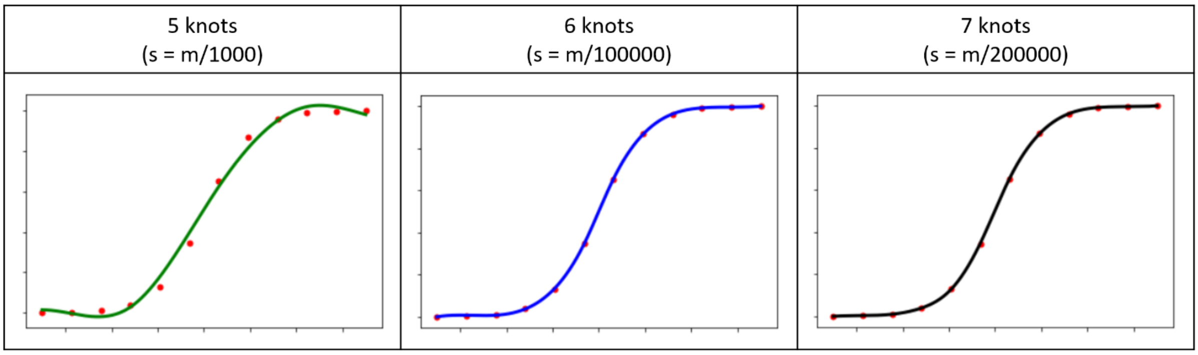


Figure S2: Generated spline for a dataset representing an S-curve trend using different smoothing factor leading to different numbers of knots.

Appendix D: Number of equidistant points for discretization

To discretize the interval of x-values for the calculation of the first and second derivatives, 100 equidistant points were chosen. In Figure 1, an example is shown of the extracted features for different numbers of equidistant points. It can be seen that going from 10 to 50 equidistant points leads to a visible difference in the width of the areas where the primitives G and E are selected by the code. Increasing the number equidistant points to 100 leads to very minor differences which are hardly noticeable, while a further increase to 200 equidistant points leads to no additional visible changes. It was decided that for the feature extraction, it is sufficient to use 100 equidistant points, as it is not necessary to get exceptionally accurate values for the positions of the extremes of the intervals to be able to extract the features of the curve.

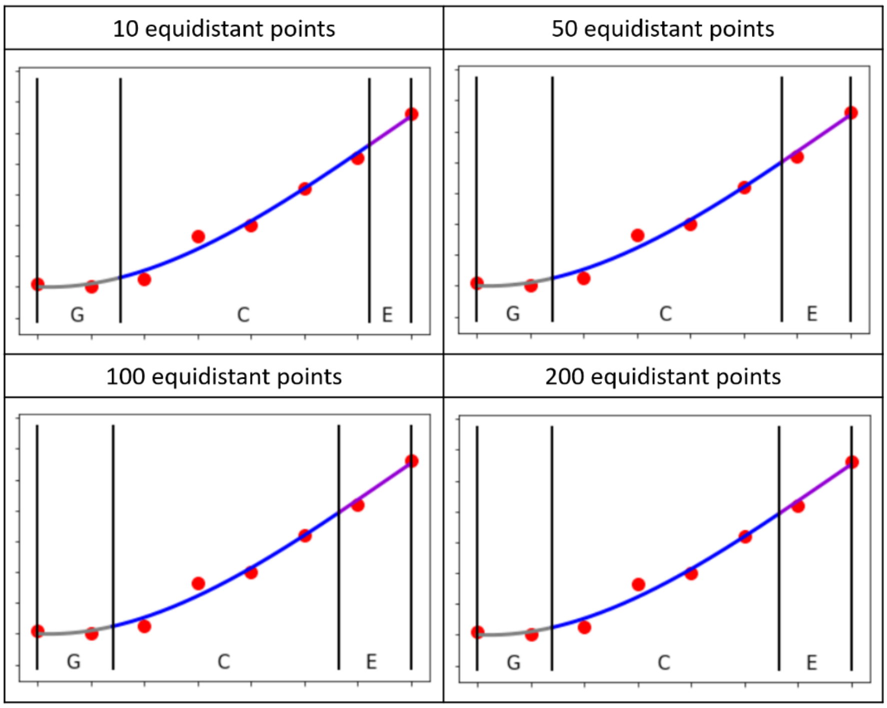


Figure S3: Extracted features from a given curve using 10, 50, 100 and 200 equidistant points for the discretization of the interval of x-values.

Appendix E: Threshold values a and b for extraction of linear trends

To extract linear trends from the data (primitives E, F and G), the values of the first and second derivatives in a certain point must be considered 0 below a certain threshold value. However, it is hard to assign a single value for all cases as the values for the derivatives depend on the units of the data and the considered ranges of x-values and y-values. To address this unit dependency, the first and second derivative are normalized by the magnitude of the respective derivative (i.e. multiplied with $\Delta x/\Delta y$ and $\Delta x^2/\Delta^2 y$ respectively to remove this dependence, , with Δx being the range of x-values, the independent variable on the abscissa, and Δy being the range of y-values, the dependent variable on the ordinate). To determine if the first or second derivative of a generated curve can be considered approximately 0, the threshold values a and b are used. The first or second derivatives are considered 0 if their absolute value is lower than a or b respectively. After several tests using fictional data, a value of 0.5 was chosen for both threshold values. Figure S3 shows the effect of the threshold values on the extracted primitives for a given curve. It can be seen that an increasing value for a leads to an increased detection of 0th order trends (primitive G), while and increasing value for b leads to an increased detection of 1st order trends (primitives E and F). Both values were chosen to be 0.5 as this led to a compromise between capturing important trends in the data while ensuring that not too many trends were automatically considered to be linear.

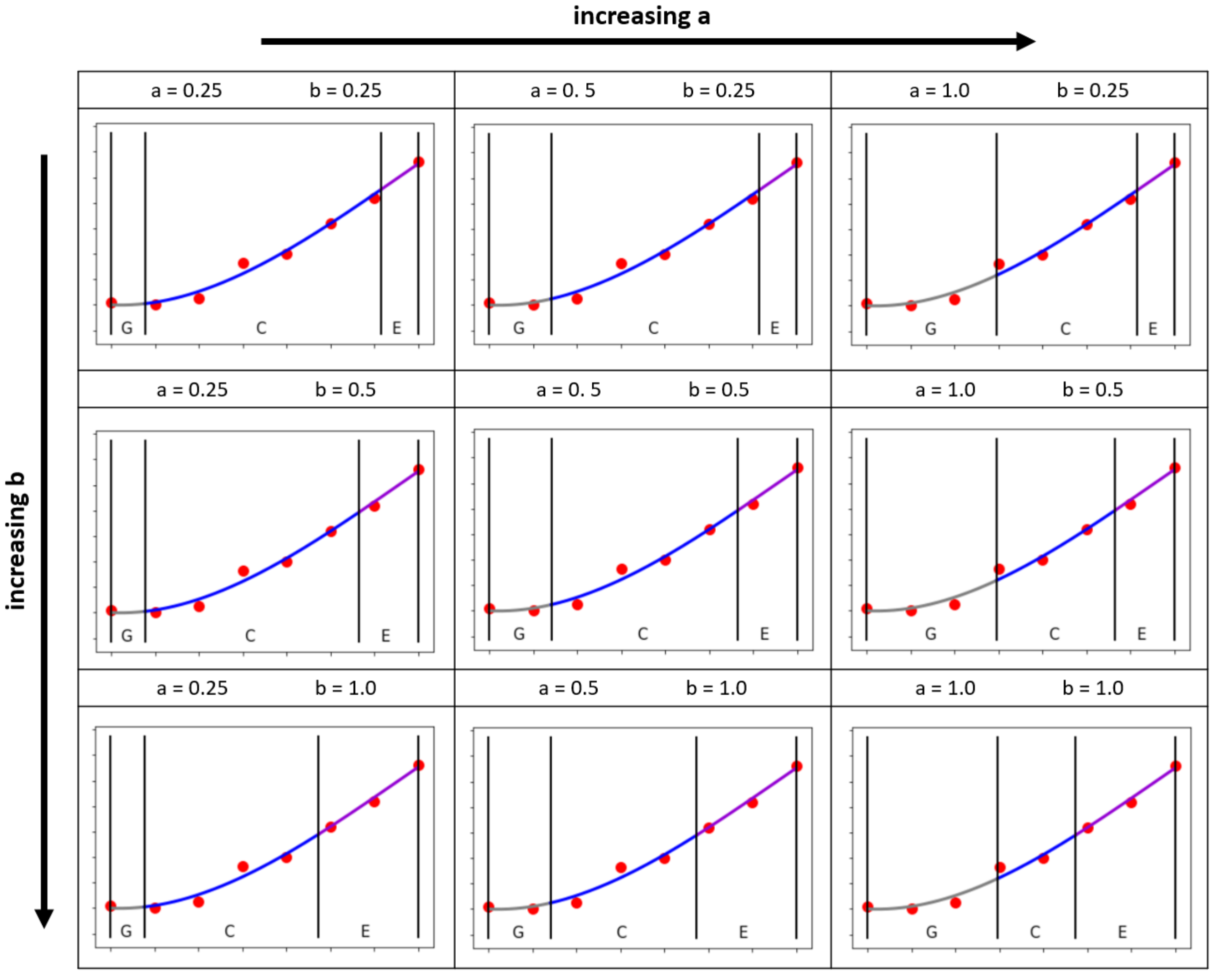


Figure S4: Extracted primitives from a given curve using different values for the threshold values a and b.

Appendix F: R-value for simplification

The multiple correlation coefficient R is defined in Eq. S6 [1].

$$R^2 = \frac{\underline{\hat{y}}^T \underline{\hat{y}}}{\underline{\hat{y}}^T \underline{\hat{y}} + \underline{e}^T \underline{e}} = \frac{S(regression)}{S(regression) + S(residuals)} \qquad (Eq.S6)$$

To compare a spline to a polynomial, the developed code compares the change in the R-value of both curves and simplifies the curve if the change is small but not when the change is large. After several tests using fictional data, it was chosen that the code would simplify the curve if the R-value of the simplified curve if larger than 99.8% of the R-value of the original curve. Some of the performed tests are shown in Figure S5. For the first dataset, the change is always lower than the threshold value, so the curve is simplified to a linear polynomial. A quadratic polynomial is selected for the second dataset, as the change between the quadratic polynomial and the cubic polynomial is small, but the change between the quadratic polynomial and the linear polynomial is considered too large. In the case of the third dataset, the spline is not simplified to a polynomial as it would lead to a large decrease in the R-value due to the fact that the cubic polynomial is not able to capture the linear trend in the data.

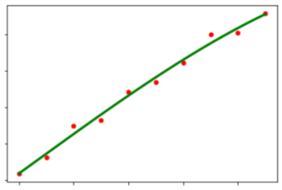
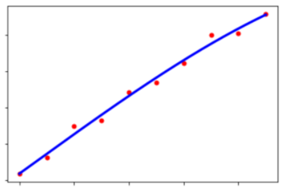
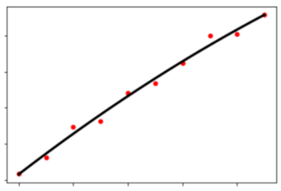
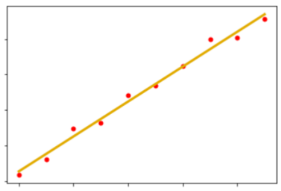
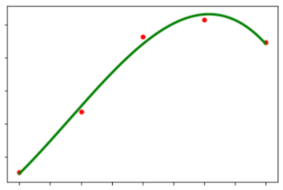
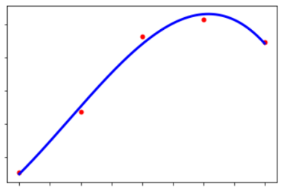
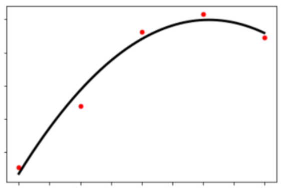
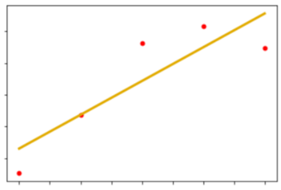
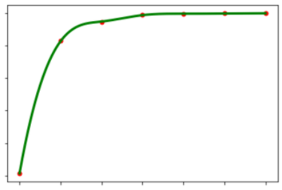
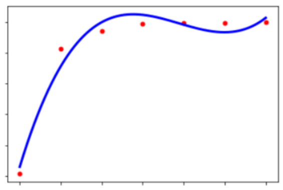
Spline	Cubic polynomial	Quadratic polynomial	Linear polynomial
R = 0.9988	R = 0.9988	R = 0.9988	R = 0.9985
			
R = 0.9993	R = 0.993	R = 0.9975	R = 0.9749
			
R ≈ 1	R = 0.9976	s = n/2000	
			

Figure S5: Generated splines and polynomials (cubic, quadratic and linear) for three datasets.

Appendix G: Logarithmic functions

When a logarithmic function is fitted to the data, it is sufficient to perform a linear regression with y in function of $ln(x)$ to obtain values for the coefficients A and B, as in Equation S7. If an exponential function is used for the regression, as in Equation S8, the logarithm must be taken of both sides of the equation to linearize it and the residual is transformed. This leads to an approximation, as a transformation of the residual is minimized and not the actual residual. Therefore, as mentioned in section 3.4.4, the logarithm is preferred.

$$y = \mathrm{A} \cdot \ln(x) + B \qquad (Eq. S7)$$

$$y = A \cdot \exp(B \cdot x) \qquad (Eq. S8)$$

The following two functions are used to fit the logarithmic functions, regardless of the x-values of the data:

$$y = \mathrm{A} \cdot \ln(x - C) + B \qquad (Eq. S9)$$

$$y = \mathrm{A} \cdot \ln\big(-(x - C)\big) + B \qquad (Eq. S10)$$

Due to the constant C introduced in the equations, it is not possible to estimate all coefficients in the equations using linear regression. The coefficient C in Eq. S9 and Eq. S10 can be estimated using Eq. S11 and Eq. S12 respectively. In these equation, x_1 and x_3 represent the smallest and the largest x-value respectively, and x_2 is the x-value of a data point between these two, which can be defined by Eq. S13. The y-values y_i correspond to the y-values of the data points with x-value x_i. To derive Eq. S9, it was assumed that $x_3 - C$ is much larger than $x_1 - C$, which means that the value for C is relatively close to the the smallest x-value x_1. For Eq. S10, on the other hand, it was assumed that $C - x_1$ is much larger than $C - x_3$, which means that the value for C is relatively close to the the largest x-value x_3. In general, these assumptions mean that the considered curves have a very “sharp” shape, which, in other words, corresponds to a fast change in the progression of the curve. These assumptions are justified, as the use of logarithmic functions was introduced for this type of progressions. Curves with much slower changes in the progression can be described just as well by polynomials, which were considered as well in the tool.

$$C = \frac{x_3 - x_1 \exp\left(\frac{y_3 - y_1}{y_2 - y_3} \cdot \ln(\alpha)\right)}{1 - \exp\left(\frac{y_3 - y_1}{y_2 - y_3} \cdot \ln(\alpha)\right)} \qquad (Eq.S11)$$

$$C = \frac{x_1 - x_3 \exp\left(\frac{y_1 - y_3}{y_2 - y_1} \cdot \ln(1 - \alpha)\right)}{1 - \exp\left(\frac{y_1 - y_3}{y_2 - y_1} \cdot \ln(1 - \alpha)\right)} \qquad (Eq.S12)$$

$$x_2 = x_1 + \alpha \cdot (x_3 - x_1) = (1 - \alpha) \cdot x_1 + \alpha \cdot x_3 \qquad (Eq.S13)$$

To estimate a value for C for any dataset, the formula is applied to all datapoints with an x-value x_2 between x_1 and x_3. The final estimation is then the average of all calculated values for C. This value is only an estimation for C and is not exactly the optimal value for C to make the sum of squares of the residuals as small as possible. Nevertheless, it is deemed more than adequate for the purpose of this work.

References

[1] Encyclopedia of Measurement and Statistics, SAGE Publications, Inc., Thousand Oaks Thousand Oaks, California, 2007.