**Application description**

The application consists in a graphical user interface that automatically calculates the binding kinetic constants of three model proteins, H-IgG, PSA, and Anti-RBD. These constants are computed to determine the affinity for their respective antibodies.

The operation of the application is described in the following four steps:

**First step:**

The user loads a .xlsx or .csv file, using the **Load data** button, with the Rtp signals, which were previously inverted using the equation:

**Rtp = 1 – Rt**,

where:

**Rt** are the original signals from the fluorescence quenching measurements caused by its interaction with graphene oxide. These signals are normalized using , where is the initial fluorescence intensity and is the fluorescence intensity obtained at each reading time.

Next, the application shows in the left widget a statistical analysis displaying the **average** (**mean**), the **standard deviation** (**std**), the **minimum value** (**min**), the **Q1 quartile** (**25%**), the **Q2 quartile** (**50%**), the **Q3 quartile** (**75%**), and the **maximum value** (**max**) of each of the **Rtp** measurements. In addition, the application generates a text file inside the output folder with this statistical information of the signals.

Figure 1 shows how the information must be organized to be used by the application. In the first column, values of time in seconds must be entered. Then, the following columns contain the measurements used to determine the kinetic constants, with m measurements. The app will automatically detect them without the need to specify. The application ignores the first row, but it is recommended to set the name of columns, time and measurements to better document the data. The penultimate row has information on the concentrations of the analytes in Molar units. The last row contains the concentrations of the analyte used. The file can contain n rows. The application will automatically detect them without the need to specify. Therefore, each column includes the signal values for each time measurement and the respective analyte concentration.

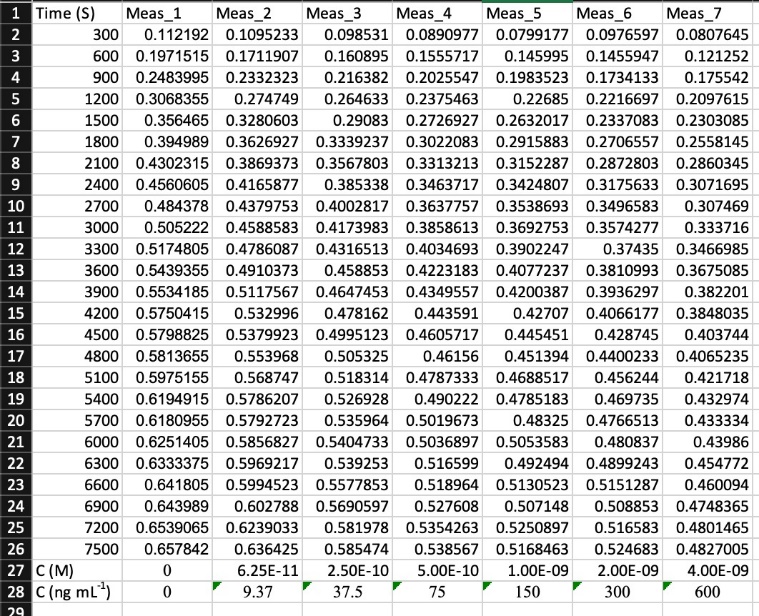


Figure 1. How the information must be organized for the determination of binding constants.

**Second step:**

In the next step, when the user presses the **Calculate button**, the application computes the calibration curves of the Rtp signals or finds a mathematical model that fits the signal measurements. Given the exponential nature of the Rtp signals, the one-phase association model (Motulsky and Christopoulos, 2004) was used:

where:

**R** are the **Rtp** signals expressed in arbitrary units,

**t** are the measurements in minutes of the **Rtp** signals,

**Plateau** is the value of **R** as it approaches infinity, expressed in the same units as **R**,

**K** is the rate constant, expressed as the reciprocal of the units of time **t**,

is the value of **R** at time 0 expressed in the same units as the **Rtp** signals.

Figure 2 displays the calibration curve using the one-phase association model with the previously mentioned parameters and variables.

Chart, line chart

Description automatically generated

**Figure 2**. Calibration curve for a single **Rtp** signal. The orange dashed graph belongs to the **Rtp** signal obtained using one-phase association model. The solid orange graph belongs to the calibration curve of the **Rpt** signal. The green dashed graph is the Plateau of the signal.

The optimization algorithm used by the application to fit the curves and determine the variables **Plateau**, **K**, and is the Lavender-Marquardt algorithm (Gavin, 2019), implemented in the SciPy library (Virtanen et al., 2020), which iteratively adjusts these variables to reduce the least-squares function:

where:

**β** are the arguments, variables **Plateau**, **K**, and , of one-phase association function . Furthermore, is the number of parameters of the measurements.

Once these variables have been calculated, the application displays on the right widget a figure with the calibration curves in continuous lines and dashed lines for the Rtp signals and saves a .png image inside the output folder. In addition, the application displays on the left widget the paramters obtained from the one-phase association model and the R2 metric of the predictions regarding the Rtp signals using the scikit-learn library (Pedregosa et al., 2011), then the application saves these values in the text file.

**Third step:**

After the **Plateau**, **K**, and variables have been obtained from all the signals, the next step of the application calculates the **Span** and **Rmax** variables.

where:

**Span** is a vector composed of the difference between and **Plateau** signals, and

**Rmax** is the maximum value of the **Span** vector.

Computed as follows:

y

**.**

The application then calculates the kinetic binding constants as follows:

where:

is the association constant (M-1s-1),

is the dissociation constant (s-1),

is the equilibrium dissociation constant (M),

is the concentration of the analyte (M).

**Fourth step:**

The application computes the absolute error to determine the relative error of the measurements using the equations:

where:

is the absolute error of the measurements,

is the relative error of the measurements,

**A** are the measurements,

is the average of measurements **A**,

is the number of measurements.

The relative error of the measurements is expressed in parentheses using a plus-minus symbol **±** in the graphical interface.

Finally, this information is displayed in the left widget and saved in the text file, along with the time in seconds that the process of determining the binding kinetic constants required.

**References**

Gavin, H.P., 2019. The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems.

Motulsky, H., Christopoulos, A., 2004. Fitting Models to Biological Data Using Linear and Nonlinear Regression: A Practical Guide to Curve Fitting. Oxford University Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. J. Mach. Learn. Res. 12, 2825–2830.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, \.Ilhan, Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, 2020. Nat. Methods 17, 261–272.